DISSERTATION

submitted

to the

Combined Faculty for the Natural Sciences and Mathematics

of

Heidelberg University, Germany

for the degree of

Doctor of Natural Sciences

Put forward by

Jannik Strötgen (MA)

born in Leonberg, Germany

Oral examination: .....

Domain-sensitive Temporal Tagging for Event-centric Information Retrieval

Advisor: Prof. Dr. Michael Gertz

Abstract

Temporal and geographic information is of major importance in virtually all contexts. Thus, it also occurs frequently in many types of text documents in the form of temporal and geographic expressions. Often, those are used to refer to something that was, is, or will be happening at some specific time and some specific place – in other words, temporal and geographic expressions are often used to refer to events. However, so far, event-related information needs are not well served by standard information retrieval approaches, which motivates the topic of this thesis: *event-centric information retrieval*.

An important characteristic of temporal and geographic expressions – and thus of two components of events – is that they can be normalized so that their meaning is unambiguous and can be placed on a timeline or pinpointed on a map. In many research areas in which natural language processing is involved, e.g., in information retrieval, document summarization, and question answering, applications can highly benefit from having access to normalized information instead of only the words as they occur in documents.

In this thesis, we present several frameworks for searching and exploring document collections with respect to occurring temporal, geographic, and event information. While we rely on an existing tool for extracting and normalizing geographic expressions, we study the task of temporal tagging, i.e., the extraction and normalization of temporal expressions. A crucial issue is that so far most research on temporal tagging dealt with English news-style documents. However, temporal expressions have to be handled in different ways depending on the domain of the documents from which they are extracted. Since we do not want to limit our research to one domain and one language, we develop the multilingual, cross-domain temporal tagger HeidelTime. It is the only publicly available temporal tagger for several languages and easy to extend to further languages. In addition, it achieves state-of-the-art evaluation results for all addressed domains and languages, and lays the foundations for all further contributions developed in this thesis.

To achieve our goal of exploiting temporal and geographic expressions for event-centric information retrieval from a variety of text documents, we introduce the concept of *spatio-temporal events* and several concepts to "compute" with temporal, geographic, and event information. These concepts are used to develop a spatio-temporal ranking approach, which does not only consider textual, temporal, and geographic query parts but also two different types of proximity information. Furthermore, we adapt the spatio-temporal search idea by presenting a framework to directly search for events. Additionally, several map-based exploration frameworks are introduced that allow a new way of exploring event information latently contained in huge document collections. Finally, an event-centric document similarity model is developed that calculates document similarity on multilingual corpora solely based on extracted and normalized event information.

Zusammenfassung

In beinahe allen Kontexten spielen Zeit- und Ortsinformationen eine bedeutende Rolle. Deshalb kommen sie in Form von Zeit- und Ortsausdrücken auch häufig in Texten vor. Oft werden dort solche Ausdrücke benutzt, um auf etwas zu referenzieren, das irgendwann irgendwo stattfand, stattfindet, oder stattfinden wird – also um auf Events zu verweisen. Bis jetzt werden Event-bezogene Informationsbedürfnisse von Standardansätzen des Information Retrievals jedoch bei weitem nicht hinreichend abgedeckt, wodurch das Thema der vorliegenden Arbeit motiviert wird: *Event-zentriertes Information Retrieval.*

Eine wichtige Eigenschaft von Zeit- und Ortsausdrücken – und somit auch eine wichtige Eigenschaft zweier Eventkomponenten – ist, dass sie normalisiert werden können, wodurch ihre Bedeutungen disambiguiert werden. Somit können sie auf einem Zeitstrahl beziehungsweise einer Karte verankert werden. Wenn statt nur der in Dokumenten vorkommenden Wörter auch normalisierte Informationen zur Verfügung stehen, können hiervon Anwendungen vieler Forschungsbereiche profitieren. Beispiele solcher Anwendungen sind Suchmaschinen, automatische Textzusammenfassungssysteme sowie Frage-Antwort-Systeme.

In der vorliegenden Arbeit präsentieren wir einige Frameworks, mit denen Dokumentensammlungen in Bezug auf zeitliche, räumliche und Event-bezogene Informationen durchsucht und exploriert werden können. Während wir uns für die Extraktion und Normalisierung von Ortsausdrücken auf ein bereits existierendes System verlassen, wenden wir uns dem Extrahieren und Normalisieren zeitlicher Ausdrücke zu. Ein kritischer Punkt ist, dass sich bisherige Arbeiten im Bereich Temporal Tagging vor allem mit englischsprachigen Nachrichtentexten, wie zum Beispiel Zeitungsartikeln, beschäftigt haben. Allerdings ist zu beachten, dass Zeitausdrücke unterschiedlich behandelt werden müssen, je nachdem aus welcher Domäne die Dokumente stammen, aus denen sie extrahiert werden. Da wir unsere Forschung jedoch nicht auf eine Domäne und Sprache beschränken wollen, entwickeln wir HeidelTime, einen Temporal Tagger, der für verschiedene Domänen und Sprachen geeignet ist. Für einige Sprachen ist HeidelTime der einzige frei verfügbare Temporal Tagger und zudem ist er problemlos für andere Sprachen erweiterbar. Außerdem erzielt er für alle unterstützten Domänen und Sprachen Evaluierungsergebnisse, die dem aktuellen Stand der Forschung entsprechen, und legt die Grundlagen für alle weiteren Beiträge, die in dieser Arbeit entwickelt werden.

Um unser Ziel zu erreichen, in Textdokumenten vorkommende Zeit- und Ortsausdrücke für Event-zentriertes Information Retrieval zu nutzen, führen wir das Konzept sogenannter *spatio-temporal events* ein. Ebenso werden Methoden entwickelt, um mit Zeit-, Orts- und Event-Informationen zu "rechnen". Diese Konzepte werden dann genutzt, um ein Rankingansatz für zeitliches und räumliches Suchen zu entwickeln. Dieser berücksichtigt nicht nur textuelle, zeitliche und räumliche Suchanfragen, sondern auch zwei verschiedene Arten sogenannter *proximity information.* Zudem passen wir unseren Ansatz der räumlich-zeitlichen Suche so an, dass direkt nach Events gesucht werden kann. Des Weiteren werden einige Karten-basierte Suchanwendungen eingeführt, die eine neue Art und Weise der Eventexploration ermöglichen. Schließlich entwickeln wir ein event-zentriertes Modell, mit dem Ähnlichkeiten zwischen Dokumenten allein anhand extrahierter und normalisierter Eventinformationen bestimmt werden.

Acknowledgements

I would like to express my deepest thanks to all the people who supported me during my PhD study. First of all, I thank my supervisor Prof. Dr. Michael Gertz. In addition to the fact that he let me enjoy an excellent research environment at Heidelberg University, he always gave helpful advice and tremendous support. We had many fruitful discussions, published several nice papers together, and, in general, he is an superb supervisor who always takes care of the students under his guidance. I also want to thank JProf. Dr. Simone Paolo Ponzetto for his reliable support, as well as the further committee members, JProf. Dr. Heike Leitte and PD Dr. Wolfgang Merkle.

During my PhD study, I had a great time at the Database Systems Research group, in particular due to a bunch of nice people. Special thanks to Conny Junghans and Natalia Ulrich who were already there when I joined the small group in 2009. Many thanks to Christian Sengstock who came shortly after me and who is a great colleague and an ideal office mate. Also many thanks to all further group members who joined step by step: Florian Flatow, Ayser Armiti, Le Van Quoc Anh, Tran Van Canh, Hamed Abdelhaq, Katarina Gavrić, Thomas Bögel, and Hui Li. Many thanks to our HiWis who helped a lot during that time, with a special thank to Julian Zell who did and does a great job in the HeidelTime project. Lunchtimes were often fun not only due to my direct colleagues but also due to the members of the Discrete and Combinatorial Optimization Group and the Parallel and Distributed Systems Group.

In addition, I want to thank those people from other institutes with whom I had the pleasure to work: the heureCLÉA members from the University of Hamburg (Prof. Dr. Jan Christoph Meister, Evelyn Gius, Marco Petris, and Janina Jacke); Omar Alonso and Ricardo Baeza-Yates with whom I had the honor to publish on temporal information retrieval; many nice people I met on conferences and had fruitful discussions with such as Leon Derczynski, Ricardo Campos, and Steven Bethard. Of course, I also thank all the people who contributed to HeidelTime either by providing helpful feedback or by developing additional language resources. They made HeidelTime much more valuable than it would have been possible without their help. Although not directly involved during the time of my thesis, I also thank Juliane Fluck for initially teaching me how to work and think scientifically during my time as student assistant at the Fraunhofer SCAI as well as Roman Klinger who helped Juliane to succeed and me to become the latex expert in my new group – in the eyes of my colleagues.

Besides the thanks within the research community, I want to thank my family and friends who believed in me during all the years. A very special thank you is dedicated to Simone. While lovely supporting me during all the years, she also made sure that there is a life next to my "PhD life" with hobbies, sports, and fantastic travels – this guaranteed that I never reached the point of loosing to enjoy the work towards finishing my PhD study. Thank you, Chérie!

Contents

1 Introduction

Machine-readable text documents can be found everywhere, and the amount of such data is increasing ad infinitum. Not only those types of textual documents that have existed for many centuries such as books, letters, and newspapers, but also rather new types of textual data such as emails, conference proceedings, Wikipedia articles, blog entries, and tweets are published continuously. In addition to texts that are available on the Internet anyway, other text types are more and more often digitized, e.g., old print books or other historic documents. Thus, it is obvious that almost every human being – at least most of the three billion Internet users[1] – is often faced with an *information overload* and requires some kind of automated help. For instance, searching the Internet for the phrase "information overload" using the well-known search engines of Google, Yahoo!, and Bing, results in over two million retrieved documents.[2]

Compared to the total number of documents on the Internet, the result set is nicely narrowed down and fortunately, documents are not returned as unsorted set but ranked according to their relevance. Thus, independent of a user's information need, search engines retrieve documents that are likely to be relevant and order them by relevance. While the standard way of using a search engine is to formulate a textual query containing words that represent an information need, the relevance of documents is determined by today's search engines on the basis of diverse information. In addition to the content of the documents that is compared to the query terms, further information is exploited, e.g., the importance of a Website – a key idea of the popular PageRank algorithm (Brin and Page, 1998) Google's search engine is based on.

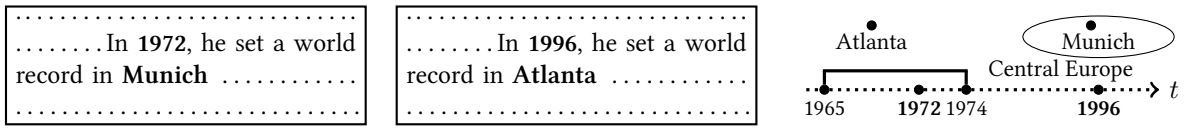
The motivation to the topic of this thesis is also grounded in the area of information retrieval. While we focus on the content of documents, we do not address textual content in general but concentrate on specific information nuggets that are important and occur frequently in many types of documents. Two of the key concepts of this thesis are thus *space* and *time*. Both share important key characteristics and play a crucial role not only in this thesis but in any information space.

1.1 Motivating Spatio-temporal and Event-centric Information Retrieval

Temporal and geographic information needs are ubiquitous. As was shown in query log analyses, many queries sent to Web search engines contain temporal or geographic terms (Nunes et al., 2008; Metzler et al., 2009). Furthermore, in many types of documents, temporal and geographic expressions are frequently used to refer to points in time and places on Earth. Examples are news articles that are usually published on a specific date and often inform about what was or will be happening on this or nearby days. Temporal expressions such as "today", "tomorrow", "last week", or "4th of November" can thus be commonly found in news documents. In addition, news are often categorized into local, regional, or global content which nicely indicates that different locations play an important role. Exactly as temporal expressions, geographic expressions, e.g., names of cities, are thus also omnipresent in news documents.

[1] According to the International Telecommunication Union (ITU) of the UN, there will be three billion Internet users by the end of 2014. `http://www.itu.int/net/pressoffice/press_releases/2014/23.aspx` [last accessed October 15, 2014].

[2] `http://www.google.com`, `http://www.yahoo.com`, `http://www.bing.com` [last accessed October 15, 2014].



(a) Document snippet A. (b) Document snippet B. (c) Required knowledge.

Figure 1.1: Motivating example showing the challenges for event-centric information retrieval.

Further examples of documents containing many temporal and geographic expressions are documents about history, e.g., when and where a battle or revolution took place, and biographies in which information about a person such as birth, death, travels, etc. can be found. Having athletes in mind, biographies could also include phrases such as the two document snippets shown in Figure 1.1.

Having a closer look on all the presented examples, temporal and geographic expressions mentioned in documents should not necessarily been considered in isolation. Often, they are used to refer to something happening at some specific time and some specific place – in other words, *events* are frequently described in textual documents. Thus, when usefully combining temporal and geographic information, event-centric search and exploration scenarios can be developed and event-centric information needs can be answered.

Independent of whether one wants to search a document collection with respect to temporal information, geographic information, or both (event information), it is important that the content of documents is not considered only as ordinary terms – as in many standard information retrieval approaches – but that temporal and geographic expressions are identified and understood as such types of information. Considering the above examples of temporal expressions, it is not sufficient that a search engine knows that a document contains the terms "today" or "last week", but it is crucial that the semantics of such expressions is accessible, e.g., that "today" refers, for instance, to the 1st of November 2014 (2014-11-01 in standard format).

Given an event-centric information need such as *"world record in Central Europe between 1965 and 1974"* and the two document snippets depicted in Figure 1.1. Then, when considering the query and the text documents as terms, both document snippets seem equally relevant. However, document snippet A can be determined as more relevant than document snippet B if the following two requirements are satisfied: (i) the search engine knows that "between 1965 and 1974" refers to a time interval, "1972" is a temporal expression, and "Munich" and "Central Europe" are geographic expressions, and (ii) the search engine is aware of temporal and geographic knowledge as depicted in Figure 1.1(c).

Generalizing this example, we claim the following: If (i) temporal and geographic expressions occurring in text documents are extracted and normalized to some standard format, if (ii) search engines provide simple ways to formulate temporal and geographic constraints in addition to a textual query part, and if (iii) knowledge about the hierarchical organization of temporal and geographic information is accessible by a search engine, then spatio-temporal and event-centric information needs can be served.

In this thesis, we address spatio-temporal and event-centric information retrieval, present spatio-temporal and event-centric search and exploration functionality, and lay the foundations to do so by developing a state-of-the-art key component for preprocessing documents, namely a tool to extract and normalize temporal expressions from text documents. In the following, we describe the challenges and our contributions in detail.

1.2 Main Challenges and Contributions

A major challenge and prerequisite for all further tasks addressed in this thesis is to extract and normalize temporal and geographic expressions occurring in documents. While we will rely on an existing tool for geographic expressions, we will develop a tool for temporal expressions, a so-called temporal tagger.

Temporal Information Extraction and the Temporal Tagger HeidelTime

Although there is quite a lot of related research on temporal information extraction, we faced the situation that existing temporal taggers were mainly developed for English news-style documents. As most tasks in natural language processing, temporal tagging is language-dependent. Less obvious but even more critical is the fact that the quality of a temporal tagger drops significantly if documents of a domain different from the original domain for which the tagger was developed are processed (Mazur and Dale, 2010). However, we neither wanted to limit our approaches to spatio-temporal and event-centric information retrieval to English as the only language nor to the news genre as the only text domain. Thus, major contributions of this thesis in the context of temporal information extraction are:

- the ***design and implementation of HeidelTime***, a temporal tagger to extract and normalize temporal expressions with high quality from documents of ***different domains and languages***,
- the development of a ***wide range of manually annotated corpora*** for languages and domains for which no temporal tagging gold standards were available so far, and
- a ***detailed analysis of the challenges and strategies*** for temporal tagging on different domains.

For the development of HeidelTime, two main challenges had to be addressed: the system architecture has to allow the simple adaptation to further languages, and it has to support multiple normalization strategies for domain-sensitive temporal tagging. HeidelTime's key features address these challenges. Its architecture strictly separates language-independent source code from language-dependent resources, and language resources can be developed following a precisely defined rule syntax. These features make HeidelTime an easy to extend multilingual, cross-domain temporal tagger.

By making HeidelTime publicly available, we make important contributions to the research community. The facts that HeidelTime achieves state-of-the-art evaluation results on all addressed domains and languages – as will be demonstrated in detailed evaluations and as we demonstrated by winning official research competitions (Verhagen et al., 2010; UzZaman et al., 2013) – and that it is the only available temporal tagger for several of its supported languages, make HeidelTime particularly valuable.

The Concept of Spatio-temporal Events

One main goal of this thesis is to exploit normalized spatio-temporal information extracted from text documents. For this, we introduce the concept of ***spatio-temporal events*** and precisely define when cooccurring temporal and geographic expressions form events. Instead of considering geographic and temporal information in isolation – as many related approaches surveyed in this thesis do – we explicitly combine temporal and geographic expressions into meaningful information nuggets.

Even though a simple cooccurrence approach to extract spatio-temporal events already results in high evaluation results, we also develop and compare several ***heuristic and linguistically-motivated approaches for spatio-temporal event extraction***.

Furthermore, we describe a concise and succinct way to usefully organize event information extracted from documents in the form of so-called ***event document profiles***, as well as temporal and geographic document profiles for extracted temporal and geographic expressions independent of whether they are parts of events. These, as well as several methods to compare temporal expressions, geographic expressions, and events with each other are the foundations for the spatio-temporal and event-centric search and exploration scenarios that will be described in the following. Note that all these concepts are developed with multilinguality in mind. They are thus applicable independent of whether the temporal, geographic, and event information was extracted from documents of one or multiple languages.

Spatio-temporal Information Retrieval

A challenge in temporal and geographic information retrieval is to provide useful querying functionality. Querying large document collections, one is usually limited to standard text search. If further querying features are provided, they are often limited to the metadata of the documents like the date and location of publication. In contrast, we exploit normalized temporal and geographic information extracted from text, and allow querying the content of the documents with temporal and geographic constraints. Thus, it is important that temporal and geographic constraints can be formulated in a meaningful way.

A further main challenge is that relevance scores for all query dimensions have to be calculated and usefully combined to allow for a meaningful ranking of documents given a multidimensional query. However, in the research areas of temporal information retrieval and geographic information retrieval – both are often considered in isolation – the dependencies between the query dimensions, i.e., text/time, text/space, or text/time/space, have been mostly ignored in the relevance ranking process.

By addressing these challenges, the contributions of this thesis in the context of spatio-temporal information retrieval are:

- a ***multidimensional query model*** to combine textual, temporal, and geographic constraints,
- the ***development of a ranking approach*** that does not only combine textual, temporal, and geographic relevance scores, but also effectively considers the ***proximity of text, temporal, and geographic expressions*** occurring in documents and satisfying the query constraints, and
- the incorporation of the ***spatial and temporal proximity of expressions to query terms*** to increase the number of ranked documents because documents not fully satisfying the temporal and geographic queries can be judged based on their distance to the query interval and region.

Furthermore, our so-called ***proximity^2-aware ranking model*** is based on index structures that allow efficient querying and retrieval of relevant documents.

Event-centric Search and Exploration in Document Collections

We already combine geographic and temporal information retrieval and consider the dependencies between query dimensions by taking into account proximity information in the context of our spatio-temporal information retrieval approach. However, when interested in an event-centric exploration of document collections, it is more intuitive to directly search for events rather than for geographic and temporal information separately, so that we additionally introduce several search and exploration frameworks for the newly introduced concept of spatio-temporal events.

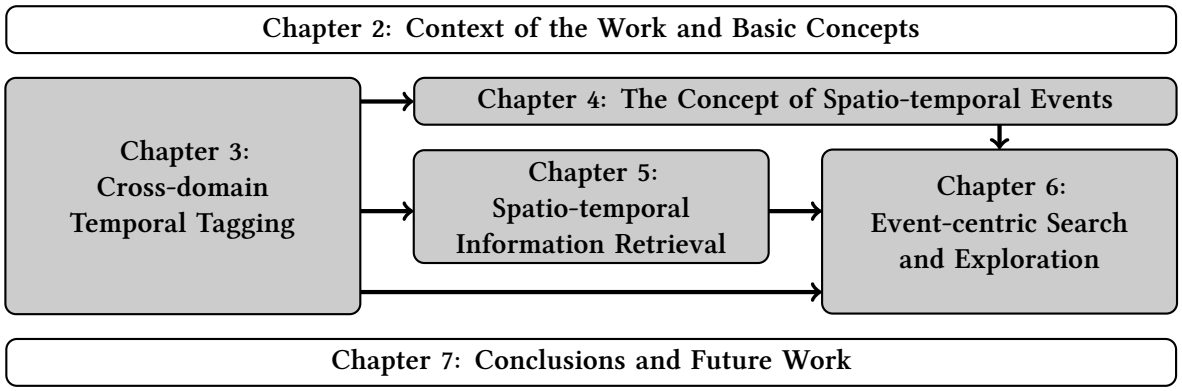


Figure 1.2: Graphical representation of the remainder of the thesis. Chapters covering main contributions are colored gray; main dependencies between these chapters are indicated with arrows.

First, we develop an ***event-centric search model*** that can be applied to either return a ranked list of relevant documents or to directly return a ranked list of relevant events. Thus, search results do not have to be explored in a per document style but can be explored on a document collection level. We furthermore describe how search results of event-centric information retrieval can be presented and explored in map-based scenarios, e.g., events of a result set can be chronologically ordered and placed on a map as an ***event sequence*** similar to trajectories studied in the context of moving object databases. This and further features lead to interesting visualization aspects in support of multilingual corpora exploration, and events are not just explorable independent of each other, but temporal and geographic relationships can directly be studied.

Another important task in the context of exploring document collections is to identify similar documents. Obviously, determining similarity is a subjective matter and documents can be similar with respect to multiple aspects. As a major contribution of this thesis and as a complement to existing similarity measures, we incrementally develop and present a ***model for an event-based document similarity measure***. While most similarity measures are based on the terms occurring in documents, our model solely relies on normalized event information. It is thus term- and language-independent and cannot only detect a non-standard type of similarity but even similarity relations across documents of different languages. In our extensive evaluation, we show the effectiveness of our model.

1.3 Outline of the Thesis

In the following, we describe the outline of this thesis. In general, the chapter descriptions are quite short since starting with Chapter 3, the four groups of contributions are covered by one chapter each. However, in addition to the content descriptions, we also explain in which of our publications parts of the content have already been published. In Figure 1.2, the structure of this thesis is also visualized graphically.

In **Chapter 2**, we place the thesis into its research context. Then, basic concepts that are of particular importance throughout the thesis are explained, e.g., the key characteristics of temporal and geographic information. In *"Temporal Information Retrieval: Challenges and Opportunities"* (Alonso et al., 2011) and in *"Event-centric Search and Exploration in Document Collections"* (Strötgen and Gertz, 2012a), we initially discussed the key characteristics of temporal and geographic information, respectively.

Important parts of the contributions of this thesis are covered in **Chapter 3**, where we first survey the research area of temporal information extraction with a particular focus on the temporal tagging task. Then, we study the challenges of cross-domain and multilingual temporal tagging. Finally, all details about our publicly available temporal tagger HeidelTime are described, e.g., its architecture, language-dependent resources, rule syntax, domain-sensitive normalization strategies, as well as a wide range of evaluations.

Since this chapter covers many contributions, several research papers contain aspects of this chapter. HeidelTime's initial development took place in the context of the TempEval-2 competition as described in *"HeidelTime: High Quality Rule-based Extraction and Normalization of Temporal Expressions"* (Strötgen and Gertz, 2010a). Its extension to cover multiple languages and different domains was first outlined together with a description of the WikiWarsDE corpus in *"WikiWarsDE: A German Corpus of Narratives Annotated with Temporal Expressions"* (Strötgen and Gertz, 2011). In the journal paper *"Multilingual and Cross-domain Temporal Tagging"* (Strötgen and Gertz, 2013a), we surveyed – although shorter than in this thesis – the state-of-the-art of temporal tagging, and explained HeidelTime's development and extension to address multiple domains and languages in detail.

The analysis of challenges and strategies for temporal tagging documents of different domains and the cross-domain evaluation also presented in Chapter 3, were initially published in *"Temporal Tagging on Different Domains: Challenges, Strategies, and Gold Standards"* (Strötgen and Gertz, 2012b). Finally, our TempEval-3 participation is covered in *"HeidelTime: Tuning English and Developing Spanish Resources for TempEval-3"* (Strötgen et al., 2013), and the journal paper *"Time for More Languages: Temporal Tagging of Arabic, Italian, Spanish, and Vietnamese"* (Strötgen et al., 2014a) explains how to extend HeidelTime to further languages, and covers challenges for temporal tagging the respective four languages.

In **Chapter 4**, we introduce the concept of spatio-temporal events. In addition, important concepts for the following chapters are developed, in particular temporal, geographic, and event document profiles as well as several functions to "compute" with event information. Partially, these concepts have been introduced in *"Extraction and Exploration of Spatio-temporal Information in Documents"* (Strötgen et al., 2010), in *"An Event-centric Model for Multilingual Document Similarity"* (Strötgen et al., 2011), and in *"Event-centric Search and Exploration in Document Collections"* (Strötgen and Gertz, 2012a).

Our approach to spatio-temporal information extraction is introduced, explained, and evaluated in **Chapter 5**. While this approach was originally presented in *"Proximity^2-aware Ranking for Textual, Temporal, and Geographic Queries"* (Strötgen and Gertz, 2013b), the multidimensional query model also detailed in this chapter, was introduced in *"Event-centric Search and Exploration in Document Collections"* (Strötgen and Gertz, 2012a).

The spatio-temporal ranking approach is adapted to event-centric search in **Chapter 6**, where event-centric search and exploration scenarios are discussed and developed. In addition, to multiple map-based techniques, we introduce the model for event-centric document similarity. Some map-based exploration scenarios were published in *"Extraction and Exploration of Spatio-temporal Information in Documents"* (Strötgen et al., 2010) and in *"TimeTrails: A System for Exploring Spatio-temporal Information in Documents"* (Strötgen and Gertz, 2010b), and have been extended in *"Event-centric Search and Exploration in Document Collections"* (Strötgen and Gertz, 2012a). Finally, the document similarity model was introduced in *"An Event-centric Model for Multilingual Document Similarity"* (Strötgen et al., 2011).

A summary of the thesis, conclusions, and suggestions for future research are presented in **Chapter 7**.

2 Context of the Work & Basic Concepts

In this chapter, we describe the context of this work and introduce several basic concepts that are elementary for this thesis. For this, we first briefly clarify in Section 2.1 the research fields that are addressed in this work, namely natural language processing, text mining, information extraction, and information retrieval. Then, in Section 2.2, named entity recognition and normalization is introduced as a first concept, which plays a central role in this thesis.

Section 2.3 addresses the concept of "time". In particular, the key characteristics of temporal information will be discussed, which motivate the importance of the temporal aspect in this work. Similarly, Section 2.4 addresses geographic information. Temporal information and geographic information are the two core concepts of our event model that will be developed in Chapter 4.

In Section 2.5, we briefly present the UIMA framework that we use for many implementations related to this thesis so that we refer to UIMA multiple times throughout the thesis. Finally, several evaluation measures will be surveyed and explained in Section 2.6 since they are crucial for both, commenting on the quality of related work approaches but also to evaluate the tools developed in the course of this thesis.

2.1 Context of the Work

Generally speaking, this thesis deals, amongst others, with processing text documents automatically. Thus, natural language processing could be considered as one broad research area of this work. However, in the context of processing text documents automatically, several terms for related and similar research fields exist. Section 2.1.1 aims at clarifying these terms and points out differences between them.

More precisely, the main goal of this thesis is to identify specific types of information in text documents, extract these types of information and make the meaning of the extracted information accessible for different automated search and exploration tasks. Thus, two more fine-grained research areas of this work are information extraction and information retrieval. These research fields will be introduced in Section 2.1.2 and Section 2.1.3, respectively.

2.1.1 Natural Language Processing and Text Mining

Natural language processing is the "engineering domain" of *computational linguistics* (Clark et al., 2010a: p.1), i.e., of "the branch of *linguistics* in which the techniques of *computer science* are applied to the analysis and synthesis of [natural] language and speech" according to the Oxford dictionary[1] definition. Thus, natural language processing is a quite broad field, e.g., considering textual but also spoken data, and this thesis only touches some topics within this large field of research.

[1] `http://www.oxforddictionaries.com/definition/english/computational-linguistics` [last accessed April 8, 2014].

Another term frequently used in the context of research related to this thesis is *text mining*. Its goal is to use natural language processing together with "techniques from data mining, machine learning, [...] information retrieval, and knowledge management" (Feldman and Sanger, 2007: Preface) to discover and extract "new information" from unstructured data (Hearst, 1999). Text mining can thus be distinguished from *data mining* since "data mining assumes that data has already been stored in a structured format" (Feldman and Sanger, 2007: p.1), e.g., in database tables, and is thus easily accessible by a computer. Information in text "is presented in an unstructured format that is not immediately suitable for automatic analysis by a computer" (Weiss et al., 2005: p.129). However, text and data mining have in common that they aim at discovering new, "heretofore unknown information" (Hearst, 1999).

In contrast, in *information extraction* and *information retrieval* applications, usually no new information is discovered but existing information is extracted from text documents and existing documents are retrieved from document collections, respectively. While this thesis also touches aspects of text mining, in particular in Chapter 6, we first address issues in the areas of information extraction (Chapter 3 and Chapter 4) and information retrieval (Chapter 5). Thus, we describe these research fields in more detail.

2.1.2 Information Extraction

Following Weiss et al. (2005), *information extraction* is "a restricted form of full natural language understanding, where we know in advance what kind of semantic information we are looking for" (Weiss et al., 2005: p.129). Typical tasks in the field of information extraction are amongst others name extraction, entity extraction, and relation extraction (Grishman, 2010: p.517).

Grishman (2010) specifies name extraction as the task to detect terms in texts as referring to persons, organizations, or locations, for instance. Entity extraction, in contrast, is specified as extracting phrases referring to specific types of entities and linking all phrases referring to the same instance of an entity (Grishman, 2010: p.517). Together, the two tasks are usually considered as *named entity extraction and normalization*, which will be further detailed in Section 2.2.

Relation extraction aims at detecting previously specified relations between entities (Grishman, 2010: p.517). Typical examples of relation extraction are affiliation relationships between a person and an organization (Grishman, 2010: p.523) and geo-spatial relationships between locations (Jurafsky and Martin, 2008: p.769), e.g., "is located in". The extraction of specific entities and relationships are two examples of information extraction tasks, which will play an important role later in this work.

2.1.3 Information Retrieval

In general, *information extraction* "must often be distinguished from *information retrieval*" (Feldman and Sanger, 2007: p.62) that will also be addressed in this thesis. While information extraction deals with the detection, delivery, and storage of specific pieces of information occurring in text documents, *information retrieval* is "what is more informally called 'search'" (Feldman and Sanger, 2007: p.62). When dealing with text documents and document collections, typical information retrieval systems return documents matching a (user-)specified query. However, the user is required to "locate the relevant information" in the returned documents (Feldman and Sanger, 2007: p.62).

Note that many information retrieval systems also perform information extraction in a limited way, e.g., by providing so-called result snippets. For a given query, a result list contains not only links to the

documents but also short summaries of the documents in addition to their titles. These summaries are usually created dynamically for a specific query and are an "attempt to explain why a particular document was retrieved for the query at hand" (Manning et al., 2008: p.157). Thus, standard Web search engines as Google, Yahoo!, and Bing are not only information retrieval systems but also apply information extraction, and, in general, the boundaries between information extraction and information retrieval often overlap.

2.2 Named Entity Recognition

A widely used information extraction application is named entity recognition (NER), which was first defined in the context of MUC-6, the sixth Message Understanding Conference (Grishman and Sundheim, 1995). Jiang (2012) considers named entity recognition as the "probably most fundamental task in information extraction [...] [because the] extraction of more complex structures such as relations [...] depends on accurate named entity recognition as preprocessing step" (Jiang, 2012: p.15). Since it also plays an important role in this thesis, the concept of named entities will be detailed in Section 2.2.1. Then, three subtasks addressing named entities will be explained, namely their extraction from text (Section 2.2.2), their classification (Section 2.2.3), and their normalization (Section 2.2.4).

2.2.1 Named Entities

According to Jiang (2012), a "named entity is a sequence of words that designates some real-world entity" (Jiang, 2012: p.15). While this definition is quite strict due to the "real-world aspect", other definitions are more general. For instance, Nadeau and Sekine (2007) state that "the word 'Named' aims to restrict [Named Entities] to only those entities for which one or many rigid designators [...] stands for the referent", with "rigid designators" being something that "in any possible world [...] designates the same object" (Kripke, 1980: p.48). Simply speaking, a named entity is thus everything "that can be referred to with a proper name" (Jurafsky and Martin, 2008: p.761) although "certain natural kind terms like biomedical species and substances" (Nadeau and Sekine, 2007) are also included.

Despite the differences in the definitions of named entities, there is no doubt that there are named entities of different types. Three types that are commonly referred to as named entities and that were already included in the sixth Message Understanding Conference (MUC-6) named entity task are persons, organizations, and locations (Grishman and Sundheim, 1995). However, "the notion of named entity is commonly extended to include things that are not entities per se, but nevertheless have practical importance and do have characteristic signatures that signal their presence" (Jurafsky and Martin, 2008: p.762), e.g., temporal expressions, monetary values, percentages, and amounts of other types of units.

Usually, the expression *named entity recognition* is used to refer to the task "to identify named entities from free-form text and to classify them into a set of predefined entity types" (Jiang, 2012: p.15). Thus, instead of NER, "the combined task of finding spans of text that constitute proper names and then classifying the entities being referred to according to their type" (Jurafsky and Martin, 2008: p.761) is also referred to as NERC – named entity recognition and classification (Nadeau and Sekine, 2007).

In this thesis, named entities also play a crucial role. Mainly temporal and geographic expressions, but also person information will be used later in this work. However, while the extraction and classification of named entities are necessary, in this work, the normalization or resolution of named entities is even more important. Thus, the processing of named entities can be split into three subtasks: the extraction, the classification, and the normalization. These will be detailed in the following sections.

2.2.2 Extraction of Named Entities

The first subtask of named entity recognition is to detect named entities in text, and can thus be considered as sequence labeling task. This labeling task is addressed by different approaches. While "early solutions [...] rely on manually crafted patterns, later systems try to automatically learn such patterns from labeled data, [and] more recent work [...] uses statistical machine learning methods" (Jiang, 2012: p.16). Note that "the assigned tags capture both the boundary and the type of any detected named entities" (Jurafsky and Martin, 2008: p.763).

Features

For the extraction of named entities, several kinds of features are usually applied: word-level features, document and corpus features, and list lookup features (Nadeau and Sekine, 2007). While word-level features consider for instance the morphology and part-of-speech of a word, document and corpus features carry information about the context of words and analyze amongst others how often words occur in a document or document collection. List lookup features – with "the terms 'gazetteer', 'lexicon' and 'dictionary' [being] often used interchangeably with the term 'list'" (Nadeau and Sekine, 2007) – check whether words or multi-word terms are part of lists containing names of specific entities such as locations, organizations, or person names.

Challenges

Note, however, that the task of named entity recognition "cannot be simply accomplished by string matching against pre-compiled gazetteers because named entities of a given entity type usually do not form a closed set" (Jiang, 2012: p.15), and "the ability to recognize previously unknown entities is an essential part of NERC systems" (Nadeau and Sekine, 2007). For some types of named entities, this ability is more important than for other types. While the amount of person and organization names is probably infinite, ways to refer to points in time are rather limited, for instance. In addition, whenever the normalization, i.e., the determination of the meaning, is the main goal of processing named entities, then using gazetteers is a promising approach.

However, recognizing previously unknown entities is not the only challenge for named entity recognition systems, and they are faced with further challenges, mainly due to ambiguity reasons: "The same name can refer to different entities of the same type, [...] identical named entity mentions can refer to entities of completely different types" (Jurafsky and Martin, 2008: p.763), and, furthermore, terms referring to named entities may also occur as regular terms not referring to any entity at all. An example of the first two ambiguity issues is the term "Washington" that can refer to different places (e.g., Washington, DC and Washington State[2]) and also to different persons (e.g., George Washington and Denzel Washington[3]). Examples for terms that can be used as regular words but also to refer to named entities are all kinds of surnames. For example, 13 of the 50 most frequent surnames in the US also occur as regular words in English: Smith, Brown, Miller, White, Lee, Walker, Hall, Young, King, Green, Baker, Hill, and Carter.[4] To handle such ambiguities, systems have to make use of more features than just list lookup features.

[2] `http://en.wikipedia.org/wiki/Washington,_D.C.`, `http://en.wikipedia.org/wiki/Washington_(state)` [last accessed April 8, 2014].

[3] `http://en.wikipedia.org/wiki/George_Washington`, `http://en.wikipedia.org/wiki/Denzel_Washington` [last accessed April 8, 2014].

[4] According to the Census 2000 of the United States Census Bureau, `http://www.census.gov/genealogy/www/data/2000surnames/Top1000.xls` [last accessed April 8, 2014].

Different Languages and Domains

As for most natural language processing tasks, the language most frequently addressed in research on named entity recognition is English. However, "language independence and multilingualism problems" (Nadeau and Sekine, 2007) are also addressed. While some features to detect named entities work well for several languages, other features are less useful for other languages. For instance, "capitalization is a good predictor of NERs in English, where common nouns are not capitalized [while in] German [...] all nouns are capitalized, but most of them are not NEs" (Faruqui and Padó, 2010).

Similarly, different domains pose different challenges. However, "the impact of textual genre [...] and domain [...] has been rather neglected in the NERC literature [...] [although] porting a system to a new domain or textual genre remains a major challenge" (Nadeau and Sekine, 2007). In this work, we will deal with multilingual and cross-domain natural language processing and will thus explain language- and domain-dependent challenges later in this work in more detail.

2.2.3 Classification of Named Entities

While the task of classification is often directly performed together with the extraction, there are different ways how named entities are classified. The typically used broad categories are persons, organizations, and locations as well as temporal (date and time) and numerical (money and percent) expressions (Sekine and Nobata, 2004), but there are also more fine-grained type specifications. For instance, subtypes of location named entities are city, state, country, etc., and fine-grained person and organization categories are also sometimes used, e.g., "politician" and "entertainer" also appear in the literature (Nadeau and Sekine, 2007). If specific domains are addressed, other types of named entities are used, e.g., genes and proteins in the biomedical domain (Park and Kim, 2006: p.124). Finally, named entity hierarchies have also been defined, e.g., the one of Sekine and Nobata (2004), which contains about 200 categories "to cover most frequent name types and rigid designators appearing in a newspaper" (Nadeau and Sekine, 2007).

In this work, temporal expressions and locations play an important role, and subtypes of these entities will be crucial. However, due to their importance in this thesis, temporal and geographic information extraction will be described separately in more detail in Section 2.3 and Section 2.4, respectively.

2.2.4 Normalization of Named Entities

The extraction and classification of named entities is already important in many information extraction scenarios. However, "NER results are often difficult to use directly, due to high synonymy and ambiguity of names across documents" (Khalid et al., 2008). Thus, determining the meaning of named entities heavily boosts the usefulness of the extracted named entities for several tasks, e.g., in information retrieval (Bunescu and Pasca, 2006; Khalid et al., 2008).

Different Names for the Task

In the literature, there are several names for very similar tasks related to determining whether multiple entity mentions refer to identical entities (and to which entities): named entity normalization (e.g., Magdy et al., 2007; Khalid et al., 2008), cross-document (entity) coreference resolution (e.g., Gooi and Allan, 2004; Singh et al., 2011), named entity disambiguation (e.g., Bunescu and Pasca, 2006; Cucerzan, 2007), entity linking (e.g., Han et al., 2011), and also entity tracking, as it was called in the entity detection and tracking task at the Automatic Content Extraction (ACE) evaluations (Florian et al., 2004).

There are slight differences between the tasks described by these terms. For example, it is not necessary that entities are linked to a knowledge base in the case of cross-document coreference resolution while entity linking implicates the "linking [of] name mentions [...] with their referent entities in a knowledge base" (Han et al., 2011). Nevertheless, the terms are often used in similar contexts. For some types of named entities, there are preferred terms to refer to the task of determining the meaning of the extracted expressions. For instance, when temporal expressions are addressed, the typically used phrase is "normalization of temporal expressions" while, in the context of geographic expressions, the task is often named "toponym resolution".

Challenges

Independent of how the task is called, "[t]he goal is to pull together all mentions of the same entity across multiple documents" (Gooi and Allan, 2004). For this, two challenges have to be addressed: ambiguity and synonymy, i.e., "disambiguating different entities with shared name mentions and normalizing identical entities with different name mentions" (Magdy et al., 2007). The first challenge is very similar to word sense disambiguation (WSD), i.e., "the task of selecting the correct sense for a word" (Jurafsky and Martin, 2008: p.672). However, in the context of WSD, regular content words are addressed, and WSD does usually "not include proper noun disambiguation" (Cucerzan, 2007). The second challenge is similar to the goal of coreference resolution, i.e., "to group together different mentions of the same underlying entities" (Weiss et al., 2005: p.145). However, in coreference resolution, not only named entities are addressed but all kinds of referring expressions, e.g., pronouns.

Assigning Identifiers or Unambiguous Normalized Values

While early approaches to name normalization "focused on intra-document normalization" (Magdy et al., 2007) in the sense of coreference resolution, more recent approaches aimed at linking entities to knowledge bases such as Wikipedia, whose pages can be used as "a unique and unambiguous way of referring to the entity" (Khalid et al., 2008).

Depending on the types of named entities, assigning identifiers or unambiguous normalized values to named entities is unequally difficult. For instance, date expressions can often be normalized based on the Gregorian Calendar, and geographic expressions can typically be normalized using some latitude/longitude information. In addition, for some types of named entities, e.g., locations and biomedical entities, fairly complete knowledge bases exist. In contrast, for entities of the types person or organization, such complete resources are not available. However, for famous and well-known persons and organizations, encyclopedic resources such as Wikipedia can be exploited. Their entries can be used as unique identifier.

In general, exploiting Wikipedia's knowledge about named entities is used in several approaches since "[i]t covers a huge number of entities [...], anchor text of inter-article links allows one to identify different text strings that can be used to refer to the same entity [...], [s]o-called 'redirects' provide information about synonyms or near synonyms, [...] [and] 'disambiguation' pages list possible referents of ambiguous names" (Khalid et al., 2008). For instance, HeiNER is a multilingual lexical resource for named entity disambiguation, translation and transliteration (Wentland et al., 2008).

As mentioned above, while person name normalization will become important later in this work, temporal and geographic expressions as well as their meaning are of uttermost importance throughout this thesis and will thus be explained in more detail in the following sections.

2.3 The Concept of Time

Temporal information "clearly plays a central role in any information space" (Alonso et al., 2011) and it also plays a central role in this thesis. As described above, temporal expression are considered as a specific type of names, which are used in natural language to refer to specific entities, e.g., points in time. Due to their importance in this work, we present in this section the key characteristics of temporal information, and then clarify the necessary terminology to describe the NER tasks of extraction, classification, and normalization of temporal expressions. The so-called task of temporal tagging which comprises these three subtasks will be surveyed and addressed in Chapter 3.

2.3.1 Key Characteristics of Temporal Information

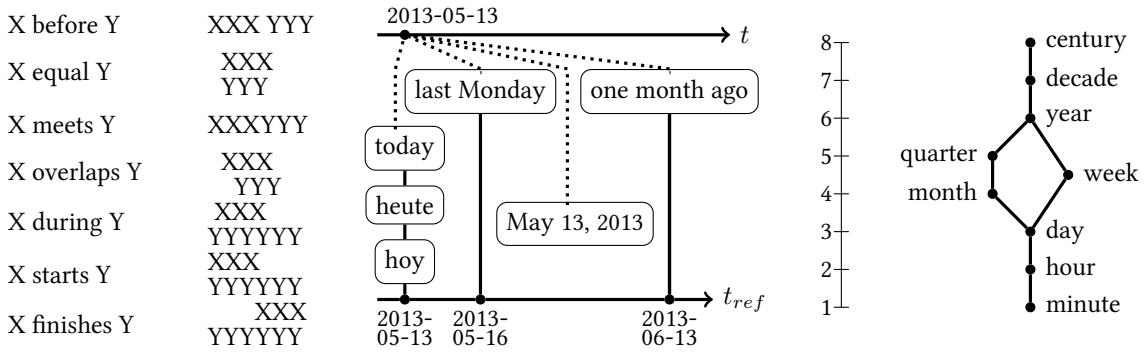
There are three key characteristics of temporal information, which make this kind of information highly valuable for many search and exploration tasks. They can be formulated as follows (Alonso et al., 2011):

- *Temporal information is well-defined*: Given two points in time or two time intervals, the temporal relationship between them can be determined, e.g., as before or identical. In general, the relationship can be assumed to be one of the temporal relations defined by Allen (1983) in the context of temporal reasoning. In addition to the equality relation, there are six symmetrical relations, namely before, meets, overlaps, during, starts, and finishes (Allen, 1983). In Figure 2.1(a), the relations are visualized following Allen's presentation.
- *Temporal information can be normalized*: Regardless of the used terms and even of the used languages, every two temporal expressions referring to the same semantics can be normalized to the same value in some standard format. Thus, temporal information can be considered as term- and language-independent. While more details on normalizing temporal expressions will be given in Chapter 3 when introducing annotation standards, an example with different temporal expressions carrying the same meaning is depicted in Figure 2.1(b).
- *Temporal information can be organized hierarchically*: Temporal information, i.e., temporal expressions carrying temporal information, can be of different granularities. For example, temporal expressions can be of granularity day (March 11, 2009), month (March 2009), or year (2009). Due to the fact that years consist of months and months consist of days, expressions of one granularity (e.g., day) can be mapped to coarser granularities (e.g., month or year) based on the hierarchy of temporal information, which is shown in Figure 2.1(c). Note that other types of granularities are also possible, e.g., seconds and millennia.

In this work, these key characteristics are crucial for several aspects, e.g., for the task of temporal tagging (Chapter 3) as well as for spatio-temporal and event-centric search and exploration tasks, which will be introduced in Chapter 5 and Chapter 6, respectively. Since the temporal information we are exploiting in this work occurs in text documents and since we will address the NER tasks of extraction, classification, and normalization, we give an overview of "temporal information in documents" in the following section.

2.3.2 Temporal Information in Documents

There are different types of temporal expressions according to what kind of temporal information an expression refers to, e.g., a point in time or a duration. In addition, there are different realizations of



(a) TI is well-defined so that one of the relations defined by Allen (1983) holds between X and Y.

(b) TI can be normalized; all expressions stated at t_{ref} have the same value in standard format (2013-05-13).

(c) TI can be organized hierarchically; mappings to coarser granularities are possible.

Figure 2.1: Visualization of the characteristics of temporal information (TI).

temporal expressions in natural language. Depending on the realization, different types of information are required to determine the normalized meaning of an expression. In this section, we introduce the categorization we will refer to in the remainder of the work, and we briefly survey how temporal expressions have been categorized in the literature.

Types of Temporal Expressions

In our work, we distinguish four types of temporal expressions:

- *Date expressions:* A date expression refers to a point in time of the granularity "day" (e.g., March 11, 2013) or any other coarser granularity, e.g., "month" (e.g., March 2013) or "year" (e.g., 2013).
- *Time expressions:* A time expression refers to a point in time of any granularity smaller than "day" such as a part of a day (e.g., Friday morning) or time of a day (e.g., 3:30 pm).
- *Duration expressions:* A duration expression provides information about the length of an interval. They can refer to intervals of different granularities (e.g., "three hours" or "five years"), and start points and end points for the interval may be determined.
- *Set expressions:* A set expression refers to the periodical aspect of an event, i.e., it describes a set of times or dates (e.g., every Monday) or a frequency within a time interval (e.g., "twice a week").

Note that date expressions – and also coarse time expressions – can also be considered as time intervals since there is always a smaller temporal unit of which such expressions consist, e.g., a single "day" as a point in time consists of hours and could thus be regarded as a duration of the granularity "hour". However, time and date expressions can always be placed on timelines as single points – although the timelines are of different granularities. In contrast, a duration expression cannot be placed on a timeline as a single point but may have a starting and an end point. Thus, time and date expressions of different granularities are not treated as durations despite the fact that they often have a duration.

In the literature, temporal expressions are sometimes categorized just into point, period, and set expressions, i.e., no explicit distinction is made between date and time expressions (see, e.g., Ferro et al., 2005b; Mazur, 2012). Furthermore, some earlier works only addressed point and period expressions (Mani and Wilson, 2000b). However, with the classification of temporal expressions into the four categories of date, time, duration, and set expressions, we follow the annotation specifications[5] of the temporal markup language TimeML (Pustejovsky et al., 2003a, 2005), which will be explained in more detail in Section 3.2.1.

Realizations of Temporal Expressions

As described in Section 2.3.1, one key characteristic of temporal expressions is that their meaning can be normalized to some standard format. However, there are different realizations of date and time expressions in natural language, which can be distinguished and which influence the difficulty of the normalization process. We distinguish four types of realizations of point expressions:

- *Explicit expressions:* Explicit expressions are date and time expressions that carry all the required information for their normalization. Thus, no further knowledge or context information is required, the expressions are fully specified and context-independent. For example, the expressions of the granularity day "March 11, 2013" and of the granularity month "March 2013" can directly be normalized to 2013-03-11 and 2013-03, respectively.

- *Implicit expressions:* Implicit expressions can be normalized once their implicit temporal semantics is known. Examples are names of holidays that can directly be associated with a point in time. A simple implicit expression is "Christmas 2013" since Christmas refers to December 25. Thus, the expression can be normalized to 2013-12-25. A more complex example is "Columbus Day 2013" since Columbus Day is scheduled as the second Monday in October. Some calculation has to be performed to normalize the expression to 2013-10-14.

- *Relative expressions:* In contrast to explicit and implicit expressions, relative expressions cannot be normalized without context information. More precisely, a reference time has to be detected to normalize expressions such as "today" and "the following year". For some relative expressions, the reference time is the point in time when the expression was formulated (e.g., for "today") while the reference time of other expressions is a point in time mentioned in the context of the expression (e.g., in the statement "in 2000 ... in the following year", 2001 is the normalized value of "the following year" since "2000" is the reference time). In both cases, the reference time is the only required information since the relation to the reference time is carried by the expressions.

- *Underspecified expressions:* For the normalization of underspecified expressions, the relation to the reference time is required in addition to the reference time itself. For instance, expressions such as "December" or "December 25" can locally be normalized without the year information. Assuming that the reference time is "November 2013" (2013-11) and the relation to the reference time is "after", then the two examples can be normalized to 2013-12 and 2013-12-25, respectively.

With this categorization, we partially follow Alonso et al. (2007) who also distinguish between explicit, implicit, and relative expressions. However, they do not separate expressions that only require a reference time for the normalization (i.e., expressions we call relative) and expressions that require a reference time and the relation to the reference time for their normalization (i.e., expressions we call underspecified).

[5] `http://www.timeml.org/site/publications/timeMLdocs/timeml_1.2.1.html` [last accessed April 8, 2014].

In general, many different names for the realizations of point expressions can be found in the literature. While the set of expressions which we call explicit expressions is usually a fixed set and only the names to refer to such expressions differ – e.g., explicit (e.g., Alonso et al., 2007; Schilder and Habel, 2001), fully specified (e.g., Pustejovsky et al., 2003a), absolute (e.g., Jurafsky and Martin, 2008; Derczynski, 2013), complete (e.g., Hinrichs, 1986), and independent (e.g., Hinrichs, 1986) – expressions we call implicit are less frequently discussed. In contrast, grouping the other expressions (i.e., the ones we refer to as relative and underspecified) resulted in different, partially overlapping sets with multiple names in the literature.

Mazur (2012) gives an overview of terminology used in the literature based on the three example expressions (i) "tomorrow", (ii) "2 days later", and (iii) "May 21st". While some authors summarize all three types of expressions, e.g., as indexical expressions (e.g., Schilder and Habel, 2001) or relative expressions (e.g., Alonso et al., 2007), they were already separated into three groups by Smith (1978) and Hinrichs (1986). Expressions such as (i) are frequently referred to as deictic expressions (e.g., Ahn et al., 2005; Busemann et al., 1997; Hinrichs, 1986; Smith, 1978). Analogously, expressions such as (ii) are referred to as anaphoric expressions by some authors (Busemann et al., 1997), while others use the same term to refer to expressions such as (ii) and (iii) (e.g., Ahn et al., 2005). Similar to our categorization, Busemann et al. (1997) name expressions such as (iii) underspecified.

Some authors include so-called *vague expressions* as a separate group of point expressions. For instance, Mani and Wilson (2000b) use the term to refer to expressions such as "Monday morning" or season names (e.g., "fall", "winter") as vague expressions since their boundaries are fuzzy, i.e., there are no exact start and end times. However, we agree with Mazur (2012) that the vagueness of such expressions should not result in a specific type of expressions since it "is not the expression that is vague [...] [but] the entity referred to that has vague boundaries" (Mazur, 2012). In addition, note that standard date and time expressions are also often used without referring to the full duration of the expression. For instance, in "he visited Germany in 2010", it is rather unlikely that the visiting took place the full year. The exact point or period in 2010 is not known, i.e., fuzzy. Thus, all expressions of a larger granularity than a timestamp could be regarded as fuzzy.[6]

In summary, the introduction of terminology to refer to specific types of point expressions probably started with the works of Smith (1978) and Hinrichs (1986), and since then, the used terminology varies between authors. The motivation for our categorization is that our four groups of point expressions directly reflect the differences in the difficulty of normalizing the expressions. While details about the normalization process will be presented in Chapter 3, a brief summary is presented in Table 2.1.

2.3.3 Extraction of Temporal Information from Documents

Research on temporal information extraction (temporal annotation) concerns the extraction of temporal expressions, events, and temporal relations between events and between events and temporal expressions (Verhagen et al., 2009). Thus, a prerequisite for the full task of temporal annotation is the extraction, classification, and normalization of temporal expressions occurring in text documents. Together, these three subtasks form the research area of temporal tagging, which is crucial not only for full temporal annotation but for many research tasks such as topic detection and tracking, summarization, and question answering (Alonso et al., 2011).

[6] As will be detailed later in this work (Section 5.2.1), for some applications it may be useful to consider every time and date expression as an interval and to assign so-called earliest and latest start and end times to them instead of a single value (Berberich et al., 2010).

realization	explicit	implicit	relative	underspecified
examples	March 11, 2013	Columbus Day 2013	today one year later	December Monday
required information for normalization	–	additional, non-standard temporal knowledge	reference time	reference time, relation to reference time

Table 2.1: The four different realizations of temporal expressions with examples and an overview of required information for their normalization.

Since some major contributions of this thesis fall into the research area of temporal tagging, we devote a full chapter of this work to the task of temporal tagging (Chapter 3). Thus, we will present in that chapter (i) existing annotation standards for temporal annotation, (ii) a survey of related work, (iii) a discussion on challenges and so far not addressed issues, as well as (iv) our contributions to the field.

2.4 Geographic Information

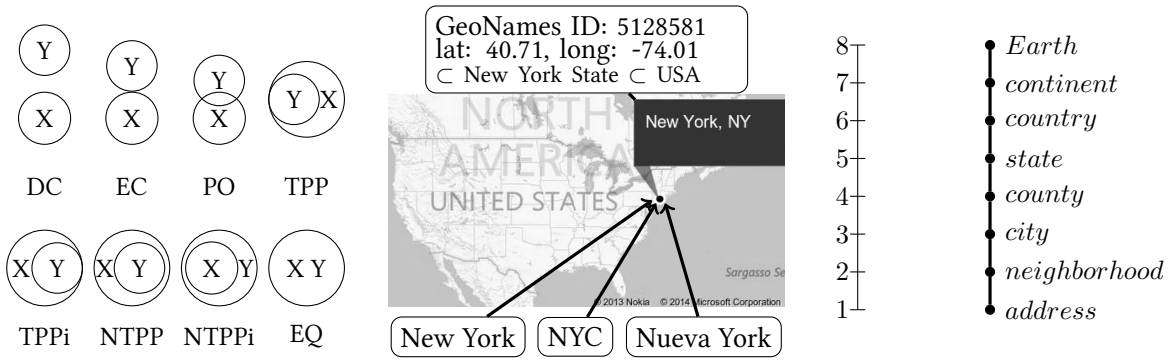
Similar to temporal information, geographic information also plays a central role in this work. Geographic expressions are the realizations in natural language to refer to a specific geographic point or region. In addition, locations have been one of the categories of named entities, which have been addressed from the early beginnings of NER research (cf. Section 2.2).

In this section, we summarize the key characteristics of geographic information and compare them with those of temporal information. Then, we describe how geographic information is realized and referred to in textual documents. Finally, we explain and survey the task of geographic information extraction with a particular focus on the extraction of geographic expressions from documents and their normalization.

2.4.1 Key Characteristics of Geographic Information

In the same way temporal expressions like dates refer to a point in time, geographic expressions refer to a location, and in general, temporal information and geographic information are very similar with respect to the key characteristics, which make them valuable for many search and exploration tasks. The key characteristics of geographic information can be formulated as follows:

- *Geographic information is well-defined:* Assuming two location points or regions, the relationship between them can be determined, e.g., as equal or overlap. For two convex regions, possible relationships are those defined in the region connection calculus RCC8 (Cohn et al., 1997). Similar to the temporal relations defined by Allen, these relations are equal (EQ), disconnected (DC), externally connected (EC), partially overlapped (PO), as well as tangential proper part (TPP) and non-tangential proper part (NTPP) and their inverses (TPPi and NTPPi, respectively). In Figure 2.2(a), the relations are visualized following Cohn et al.'s presentation.
- *Geographic information can be normalized:* Regardless of the used terms and even of the used language, every two geographic expressions referring to the same location can be normalized to the same value in standard format. This normalized information is typically a unique identifier to a knowledge base where further information, e.g., some latitude/longitude information is listed (e.g., a point, a rectangle, or a polygonal region). While further information about how



(a) GI is well-defined; one of the relations defined by Cohn et al. (1997) holds between X and Y.

(b) GI can be normalized; all expressions have the same value in standard format (latitude/longitude).

(c) GI can be organized hierarchically; mappings to coarser granularities are possible.

Figure 2.2: Visualization of the characteristics of geographic information (GI).

geographic expressions can be normalized will be provided in Section 2.4.3, an example with different expressions referring to the same location is depicted in Figure 2.2(b).

- *Geographic information can be organized hierarchically:* As temporal expressions, geographic expressions can be of different granularity. For instance, an expression can refer to a city (e.g., New York City), state (e.g., New York State), or country (e.g., USA). Since locations of smaller granularity typically lie in a larger region, which can also be referred to by a name, they can be mapped to coarser granularities (e.g., a city to a state). In Figure 2.2(c), a geographic hierarchy is presented. Note that other types of locations are also possible, e.g., points-of-interest or rivers and mountains.

In summary, temporal information and geographic information share some main key characteristics that are crucial for several aspects in this work, e.g., for the spatio-temporal and event-centric search and exploration scenarios developed in Chapter 5 and Chapter 6, respectively. However, to be able to exploit geographic expressions occurring in text documents, they have to be extracted and normalized. Before discussing the extraction and normalization of geographic expressions in Section 2.4.3, we explain in the next section how geographic expressions occur in text documents.

2.4.2 Geographic Information in Documents

Similar to temporal expressions, geographic expressions can be categorized into different types, and there are different realizations of geographic expressions in natural language to refer to a specific location.

Types of Geographic Expressions

Intuitively, geographic expressions can be categorized as location points and location regions – analogously as two of the categories of temporal expressions, points in time and temporal intervals. Then, a location point would be a location without geographic extent and would be normalized to a single latitude/longitude pair, while regions would be either associated with a rectangle (i.e., a bounding box) or a more complex polygon. However, there are some reasons why such a distinction is neither necessary nor useful in the context of our work:

- Similar to time and date expressions, points are just locations of the smallest possible granularity. However, in contrast to durations as temporal intervals, both location points and location regions are normalized according to the same information, i.e., some latitude/longitude information which can be placed on a map. In contrast, date and time expressions are normalized as points on a timeline while durations are normalized according to their length and cannot necessarily be placed on a timeline as briefly introduced in Section 2.3.

- In the context of information extraction, even locations with spatial extents are assigned some point information by most information extraction system. For example, although New York City clearly has some extent, it might be assigned only point information only as in Figure 2.2(b). However, to many locations, polygonal information can also be assigned. Whether point of region information is associated with an extracted geographic expressions is thus rather dependent on the information extraction system. Note, however, that spatial relations can nevertheless be determined if the system is aware of containment and hierarchy information of known entities as shown in Figure 2.2(b) for New York City (NYC $\subset$ New York State $\subset$ USA). Details about how geographic expressions are typically extracted from text documents and normalized to some real-world locations will be presented in Section 2.4.3.

- Finally, any location has some extent and "the space occupied by any real physical body will always be a region rather a point" (Cohn et al., 1997). Despite the name, even points of interest may have a spatial extent. Thus, we do not categorize locations in point and region locations, but assume that all locations referred to by geographic expressions are handled equally. Nevertheless, there are of course other categories for locations as will be presented in the following.

Geographic expressions often refer to locations with strict administrative boundaries (e.g., city boundaries) so that typical sub-categories of location entities are administrative regions such as city, state, and country. These administrative boundaries are important since they often directly provide containment information about locations. For instance, assuming a city l_{city}, a state l_{state}, and a country $l_{country}$, then, although an information extraction system may associate latitude/longitude information as a single point, additional information might also be available in the underlying knowledge base. For example, that l_{city} is located in l_{state}, which is again located in $l_{country}$. In summary, the geographic relationship between the three locations can be determined – either based on polygonal latitude/longitude information or based on available hierarchical containment information.

There are also geographic expressions to refer to locations without explicit boundaries, e.g., names for neighborhoods (e.g., Schockaert and Cock, 2007) or expressions such as "the south of Germany". Note however, that due to the characteristic of geographic expressions that they can be organized hierarchically, such expressions can be mapped to the next coarser granularity (cf. Section 2.4.1).

Realizations of Geographic Expressions

In contrast to temporal expressions, geographic expressions are often classical "names". Nevertheless, there are also locations without explicit names so that *relative geographic expressions* are used to refer to them. For instance, phrases such as "twenty miles south of l" are used to refer to a location that either has no name or no well-known name, but which is located south of the named location l. Obviously, only four (north, east, south, and west) or eight direction specifications (north-east, north-west, south-east, and south-west, additionally) typically occur for relative expressions. Thus, given a point location l,

expressions such as "twenty miles south of *l*" are still difficult to normalize since they refer to a rather large and fuzzy region. However, in contrast to relative temporal expressions, relative geographic expressions are usually not used in text documents if a name for the location exists – except for the purpose of providing additional information. Thus, named entity extraction for locations typically concentrates on extracting locations referred to by a name.

Similarly, although locations can be described by their normalized latitude/longitude information (either as point or as region), this typically does not occur in textual documents for the sake of readability. Thus, only names are typically used to refer to locations. However, names are often not explicit in the sense of being unambiguous as explicit temporal expressions, and different types of ambiguities have to be considered for determining to which specific location a geographic expression refers.

Using the terminology introduced for different realizations of temporal expressions (cf. Section 2.3.2), geographic expressions could be considered as either explicit (if there is only one location carrying a specific name) or as underspecified in case of ambiguity. Since the distinction of underspecified and explicit geographic expressions would only depend on whether there are multiple entities with the same name, and not on characteristics of the expression itself, we do not use a distinction of realizations of geographic expressions analogously to the one for temporal expressions.

In lieu thereof, we distinguish between *relative geographic expressions* and *geographic expressions as location names*. The latter ones are often referred to as toponyms in the literature (see, e.g., Leidner, 2007). Toponyms are the main subject of analysis in the context of extracting and normalizing geographic expressions, which will be detailed in Section 2.4.3 after explaining different ambiguity issues of toponyms.

Ambiguities of Toponyms

The main reason why the two subtasks recognition of toponyms and resolution of toponyms are non-trivial is due to ambiguity issues. On the one hand, there are many different locations with identical names as exemplarily shown in Figure 2.3 for some of the locations in the eastern part of the United States called "Springfield". On the other hand, there are several location names that have another non-geographic meaning, e.g., Mobile (Alabama) vs. mobile (as common adjective or noun). These two types of ambiguities are called geo/geo ambiguities and geo/non-geo ambiguities, respectively (Amitay et al., 2004). Note that geo/non-geo ambiguities can either occur if a location name can also refer to another entity type such as person (e.g., Washington (D.C.) vs. George Washington) or if a location name is a common word. In Table 2.2, we list several examples of geo/geo and geo/non-geo ambiguities.

Note that these polysemy issues are not the only types of ambiguities related to toponyms, but there are also synonymy issues, i.e., a location can be referred to using different names. For instance, the terms New York City, NYC, New York, and Big Apple are all frequently used to refer to the same city. In addition, there are often differing spellings in different languages, e.g., the Spanish standard term to refer to New York is Nueva York (cf. Figure 2.2(b), page 18). Similar as for time information where expressions can refer to different times and different expressions can be used to refer to the same point in time (cf. Section 2.3.2), these ambiguities of toponyms have to be considered if the tasks is to extract and normalize geographic expressions from documents as will be explained next.

Figure 2.3: Some of the cities and towns in the Eastern United States called "Springfield". Source: `http://www.geonames.org/maps/showOnMap?q=Springfield&country=US`.

	geo/non-geo ambiguities	
	Washington	(George) Washington (person)
	Jordan	(Michael) Jordan (person)
*	To, Myanmar	to (preposition)
*	Reading, England, UK	reading (verb, noun)
+	In, Thailand	in (preposition)
+	Of, Turkey	of (preposition)

	geo/geo ambiguities	
	Heidelberg, Germany	Heidelberg, South Africa
	Washington, D.C., USA	Washington (State), USA
+	Boston, England, UK	Boston, MA, USA
+	Cambridge, England, UK	Cambridge, South Africa

Table 2.2: Some examples of geo/geo ambiguities and geo/non-geo ambiguities.
Sources: *Amitay et al. (2004), +Leidner (2007).

2.4.3 Extraction of Geographic Information from Documents

Research on the extraction of geographic information from documents has a long tradition. As one of three categories, locations were already addressed in the beginnings of NER research (cf. Section 2.2). However, for the exploitation of geographic information occurring in text documents, it is important to not only extract geographic expressions, but to also normalize them to some entry in a knowledge base associated with some latitude/longitude information. In contrast to the extraction and the normalization of temporal expressions, we do not address these tasks for geographic expressions but rely on an existing, publicly available tool. Since there is no separate chapter on geographic information extraction, we briefly survey in the following some research on the combined task of extraction and normalization of geographic expressions typically referred to as geo-tagging, geo-parsing, or geo-coding in the literature.

As already mentioned in Section 2.2, there are many different names for the two subtasks. The first task is often referred to as toponym recognition, extraction of geographic expressions, or just as one of the tasks in general named entity recognition. Accordingly, the second subtask is called resolution (e.g., Lieberman et al., 2010; Leidner, 2007), grounding (e.g., Leidner et al., 2003) or normalization (Li et al., 2002). Independent of the names used to refer to the tasks, there are several approaches described in the literature addressing them. In the following, we briefly explain so-called gazetteers, and present strategies for the extraction and normalization of geographic expressions.

Gazetteers

In the context of extracting and normalizing geographic expressions from documents but also in the context of all kinds of geographic information systems applications, so-called gazetteers play an important role. They can be "considered to be a type of knowledge organization system" (Hill, 2006: p.92) or knowledge base for geographic information, or more precisely, for real-world entities of the type location.

While in the research field of named entity recognition, the terms gazetteer, list, dictionary, and lexicon are often used interchangeably (Nadeau and Sekine, 2007), gazetteers are typically not just lists of names but contain a lot of additional information. Thus, they are "large lists of names of geographic entities [...] enriched with further information, such as their class [...], their size, and their location" (Leidner et al., 2003). However, typically, not the name of a location is in the center of a gazetteer, but "there should be one entry for each place" (Hill, 2006: p.93), i.e., the location entity itself is an entry in the gazetteer. Each entry is associated with further information such as alternate names, type of location (e.g., city, state, country), latitude/longitude information, but also additional information such as altitude, population size, and specific features (e.g., populated place, capital). Furthermore, many gazetteers do not only contain information about single entries, but put entries into relations resulting in hierarchy information.

There are several gazetteers available with some of them covering only specific regions, e.g., the United States, and others covering the whole Earth (with different degrees of completeness). Three examples of gazetteers are GeoNames[7], the Getty Thesaurus of Geographic Names[8], and Yahoo! GeoPlanet[9]. As will be detailed next, gazetteers play an important role not only for normalizing geographic expressions but also for their extraction.

Extraction of Geographic Expressions

Leidner and Lieberman (2011) split up current approaches to the extraction of geographic expressions into three categories:

- *Gazetteer Lookup Based:* "The text is traversed [...] and searched for occurrences of a predefined set of toponyms [...] stored in a gazetteer" (Leidner and Lieberman, 2011). Such approaches only miss to mark geographic expressions if the gazetteer is not aware of a specific toponym, i.e., if the gazetteer is not complete. However, geo/non-geo ambiguities are not resolved, which usually results in several incorrectly extracted expressions. Thus, pure gazetteer lookup-based systems can usually be regarded as recall-oriented. They are typically not used without a normalization component since the resolution of all ambiguities is left to the normalization step.
- *Rule Based:* "A set of symbolic rules in a domain-specific language encodes a decision procedure that permits an interpreter to decide whether a word is a toponym or not" (Leidner and Lieberman, 2011). These rules are typically built using regular expressions or grammars based on them. The goal is to cover typical phrases (e.g., "city of <token>") or terms with typical suffixes for toponyms (e.g., words ending with "shire"). In addition, the rules can also be formulated in a context-dependent way, e.g., that terms preceded by prepositions such as "from", "to", and "in" are candidates for toponyms if they are capitalized.
- *Machine Learning Based:* "Based on a training corpus containing [...] [manually created gold standard annotations], feature configurations that are most highly correlated with toponyms are extracted" (Leidner and Lieberman, 2011). The same features are then calculated for each position in new texts and the most likely class (toponym or non-toponym) is selected for each word. Features are often formulated in a boolean way, e.g., whether or not a word is capitalized. As mentioned in Section 2.2.2, one feature may also be whether a term is listed in a gazetteer (Nadeau and Sekine, 2007).

[7] `http://www.geonames.org/` [last accessed April 8, 2014].
[8] `http://www.getty.edu/research/tools/vocabularies/tgn/` [last accessed April 8, 2014].
[9] `http://developer.yahoo.com/geo/geoplanet/data/` [last accessed April 8, 2014].

Both, the second and the third approach are typical methods for the general task of named entity recognition (cf. Section 2.2). In contrast to pure gazetteer lookup based methods, geo/non-geo ambiguities are directly addressed, and only the resolution of geo/geo ambiguities is left to the normalization step. Thus, if the task is to only extract toponyms without normalizing them to a real-world entity, these NER methods usually achieve better results. However, if both tasks are addressed, the extraction quality of gazetteer lookup based approaches is improved since geo/non-geo ambiguities are resolved and potential toponyms are removed if they are considered as being used with their non-geographic meaning. In addition, note that wrong decisions during the extraction phase cannot be corrected in the case of rule-based and machine learning based extraction methods.

Independent of whether gazetteers were used during the extraction phase, they are required for the normalization of toponyms, which will be described next.

Normalization of Geographic Expressions

Depending on the extraction approach, the strategy for the normalization task of toponyms either has to resolve solely geo/geo ambiguities or geo/geo and geo/non-geo ambiguities. The used features are very similar although in the latter case, some features may be more complex and additional features may be included, e.g., so-called black lists containing potential toponyms frequently used without carrying the geographic meaning.

There are several approaches described in the literature for the normalization of geographic expressions, and many of them use a combination of similar features. The group of features can be separated into (i) NLP methods and (ii) world knowledge (Leidner, 2007). Frequently used NLP methods are as follows:

- *One sense per discourse:* Originally formulated by Gale et al. (1992) in the context of word sense disambiguation research, this principle states that if an ambiguous word occurs more than once in a text, "it is extremely likely that [all occurrences] will all share the same sense" (Gale et al., 1992). This principle is frequently applied for toponym resolution (see, e.g., Li et al., 2002; Leidner, 2007; Lieberman et al., 2010). Often, this principle is applied to all remaining toponyms each time an ambiguous toponym is resolved (see, e.g., Lieberman et al., 2010).
- *Local context:* The local context of potential toponyms, i.e., the surrounding tokens, can help to resolve both, geo/non-geo and geo/geo ambiguities. For instance, assuming a potential toponym is preceded by (i) "city of", (ii) "state of", or (iii) a capitalized token being in a list of standard first names, then (i) and (ii) are hints that the toponym is a city (e.g., "city of Washington") and a state (e.g., "state of Washington"), respectively, while (iii) is a hint that the potential toponym is no location at all (e.g., "George Washington") (see, e.g., Li et al., 2002).
- *Qualifying context:* In many text documents, ambiguous toponyms are directly followed by another toponym with the first one being hierarchically located in the second one (see, e.g., Leidner, 2007; Lieberman et al., 2010). That is, locations are textually disambiguated by the author, e.g., "Cambridge, MA" or "London (Ontario)".

In addition, there are several features that require some world knowledge. As mentioned above, gazetteers often contain much more information about locations than just latitude/longitude information. Examples of world knowledge features are:

- *Default sense:* An often applied strategy to ground toponyms which cannot be resolved otherwise is to determine a default sense for each toponym. For this, the location with the highest population (information often stored in gazetteers for populated places) is typically selected as default sense. In addition, sometimes countries and capitals are preferred over other locations.
- *Spatial proximity:* Toponyms are resolved in such a way that the area which is covered by the minimal bounding box or polygon of all locations is minimal (Leidner, 2007). Alternatively, the pairwise spatial proximity can also be minimized (Lieberman et al., 2007).
- *Black list:* In particular if geo/non-geo ambiguities have not been resolved during the extraction phase, a black lists comes into play. It contains toponyms which are very unlikely to refer to locations since they have a frequently used non-geographic meaning such as "In" and "Of" (cf. Table 2.2, page 21).

A more detailed overview with several additional features described in the literature is given by Leidner (2007). Of course, there are also many different ways how to combine the single features, e.g., having a feature hierarchy or calculating a final score for each toponym based on all features.

In addition, Lieberman et al. (2010) points out that it is important to use so-called local and global lexicons for resolving ambiguous toponyms. The assumption is that there are globally relevant locations (e.g., names of countries and capitals, further important and well-known cities/locations), and locations which are locally relevant. Thus, given the location where a document is published (e.g., a local newspaper), the local lexicon contains all of those locations which are in spatial proximity and thus locally relevant. In contrast, all other small and not well-known locations are excluded from the lexicons and thus no candidates for ambiguous toponyms. For example, given the toponym "Paris" in a local newspaper from Texas and in another newspaper, then the location "Paris, Texas" is part of the local lexicon in the first case but not in the second case. Thus, the ambiguity between "Paris, France" and "Paris, Texas" has to be resolved only in the first case (for more details, see Lieberman et al., 2010).

Available Geo-Taggers

Although there are many approaches described in the literature, many of them are just research prototypes and not publicly available. Nevertheless, there are a few publicly available geo-taggers such as Yahoo! Placemaker on which we also rely in our work for extracting and normalizing geographic expressions as will be described in Chapter 4.

2.5 UIMA: Unstructured Information Management Architecture

From the practical perspective, one of the goals of this thesis is to process textual documents to extract and normalize specific types of information from documents and to use the extracted information in manifold search and exploration tasks. To achieve the goal of successfully extracting and normalizing information, we will apply several natural language processing tools, e.g., for temporal tagging and geo-tagging, but also for standard linguistic preprocessing tasks such as tokenization and part-of-speech tagging. Thus, it is important to combine different tools, which have originally not been developed to be used together, in an easy and straightforward way.

A publicly available and widely used open source software architecture supporting this idea, is UIMA, the Unstructured Information Management Architecture.[10] Originally, UIMA was developed at IBM, but in 2006 it became an incubator project at Apache Software Foundation[11], and since 2010, it is a top level Apache project. In general, UIMA is not only an architecture and framework for processing textual documents, but also for other types of unstructured data, e.g., image, audio, and video. UIMA's main objective is described as "to support a thriving [research- and industry-based] community of users and developers of UIMA frameworks, tools, and annotators, facilitating the analysis of unstructured content [...] in order to discover knowledge that is relevant to an end user".[12]

In this section, we briefly explain UIMA's basics since we will refer to UIMA multiple times throughout the thesis, e.g., when presenting our UIMA kit for temporal tagging (Chapter 3), as well as when describing our text processing pipelines for event extraction (Section 4) and for making available geographic and temporal information for spatio-temporal and event-centric search and exploration tasks (Chapter 5 and Chapter 6, respectively). Although there are further frameworks such as NLTK[13] (Bird et al., 2009) and GATE[14] (Cunningham et al., 2011), in the context of this thesis, we rely on UIMA for linguistic processing.

The Pipeline Principle

UIMA-based applications are organized as a pipeline and are thus decomposed into components. Each component fulfills a specific task and multiple components can be combined with each other since they all use the same data structure (the Common Analysis Structure, CAS), and UIMA organizes the data flow between them. As will be detailed below, there are three types of components in a pipeline: Collection Readers, Analysis Engines, and CAS Consumers.

While either written in Java or C++, each component does not only consist of a programmatic part where UIMA interfaces are implemented, but also contains a descriptive part with metadata about the component itself and a description of the Type System, on which the component relies.

Common Analysis Structure and Type System

The Common Analysis Structure (CAS) is the basis of all components in a UIMA pipeline. A CAS object is initialized at the beginning of the pipeline for each document that is to be processed. It is also loaded and possibly extended by later used components. The CAS is an object-based data structure and its objects can be accessed and manipulated via UIMA's interfaces. Thus, component developers can easily add annotations by instantiating classes of the annotation type that is to be added, while UIMA controls the data representation (Ferrucci and Lally, 2004a).

While several basic annotation types are defined by UIMA, these can be extended and other annotation types can be added via a so-called Type System. The Type System, which can be thought of "as an object schema for the CAS",[15] usually defines all object types that one expects to extract from a document collection. Furthermore, subtypes can be defined via inheritance relations from other types, and features

[10] `http://uima.apache.org/` [last accessed April 8, 2014].
[11] `http://www-03.ibm.com/press/us/en/pressrelease/20625.wss` [last accessed April 8, 2014].
[12] `http://uima.apache.org/index.html` [last accessed April 8, 2014].
[13] `http://www.nltk.org/` [last accessed April 8, 2014].
[14] `http://gate.ac.uk/` [last accessed April 8, 2014].
[15] `http://uima.apache.org/d/uimaj-2.5.0/overview_and_setup.html` [last accessed April 8, 2014].

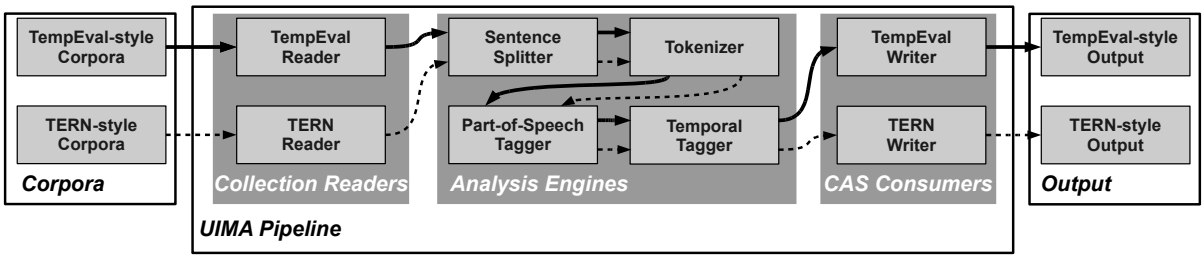


Figure 2.4: An example UIMA pipeline for temporal tagging corpora in two different formats.

can be associated to the types. A linguistically-motivated example is a type "token" containing UIMA's standard features for annotations "begin" and "end" to describe the position of the token in the document, and the additionally defined feature "POS" for part-of-speech information.

Collection Readers

A Collection Reader is the first component of a UIMA pipeline. It reads data from a source (file system, database, etc.), decides how to iterate over the documents, and initializes a CAS object for every subject of analysis. The Collection Reader may add metadata to the CAS object as well as the document text, which is the most important part for our purposes since we are processing text documents. An important idea behind using Collection Readers is to keep the remainder of the pipeline independent of the characteristics of the data source.

Analysis Engines

The Analysis Engines are the components of a pipeline that analyze a document, find information, and annotate this information in the CAS object. The first Analysis Engine of a pipeline gets the CAS object from the Collection Reader, the other Analysis Engines from the former Analysis Engine. The extracted or derived information, the Analysis Results, typically contain metadata to the content of a document. For example, a sentence splitter adds sentence boundary information to the CAS. The Analysis Engines can access all the information available in the CAS object, e.g., a later used tokenizer can access the Analysis Results of an earlier used sentence splitter.

CAS Consumers

The last components of a pipeline are the CAS Consumers. They do not add any further information to the CAS object, but do the final processing. Possible tasks are, for example, to store all analysis results in a database, to visualize specific types of annotations, or to output some annotations in a specified format, e.g., to perform an evaluation against a gold standard.

An Example UIMA Pipeline

In Figure 2.4, an example UIMA pipeline is depicted, which is motivated by the following scenario. There are temporally annotated gold standard corpora in two different formats (TempEval- and TERN-style formats), and there is a temporal tagger which shall be evaluated on the gold standard corpora. Note that there are two workflows – one for each corpus format – represented by the two arrow types.

The first components in the pipeline are the Collection Readers "TempEval Reader" and "TERN Reader". These are used to read the corpora and to instantiate a CAS object for each input document, i.e., to make available the documents' texts independent of the original formatting of the documents.

Then, the Analysis Engines are applied. First, a sentence splitter, a tokenizer, and a part-of-speech tagger perform linguistic preprocessing. Obviously, the sentence splitter uses the document text as provided by the Collection Readers and adds sentence annotations to the CAS objects, while the tokenizer splits each sentence into tokens and adds token annotations. Similarly, the part-of-speech tagger processes the sentences to add the part-of-speech feature to each token annotation. As last Analysis Engine, the temporal tagger extracts temporal expressions for which it requires token and part-of-speech information. That is, later used Analysis Engines make use of the Analysis Results of previously applied ones.

Finally, to be able to compare the temporal expressions extracted by the temporal tagger to the annotations in the gold standard corpora, the CAS Consumers "TempEval Writer" and "TERN Writer" are applied to produce the output documents in the TempEval- and TERN-style formats, respectively.

Note that all components of the pipeline except the Collection Readers are independent of the input format. For instance, the part-of-speech tagger does not care about the original format of the documents and even the CAS Consumers could be used for all documents. Thus, it would be possible to output the documents of the TempEval-style corpora in TERN-style format after processing the documents.[16]

2.6 Evaluation Measures

In the following chapters, information extraction and information retrieval methods will be surveyed and developed. To measure the quality of such methods, evaluations are performed by comparing a system's output to some ground truth and by calculating specific evaluation measures. Using standard evaluation metrics also allows to compare different systems with each other. In this section, we briefly explain all of those evaluation measures frequently used in information extraction and information retrieval, which will be used and referred to throughout this work when presenting evaluation results of existing tools and our newly developed methods. Section 2.6.1 and Section 2.6.2 cover evaluation measures for information extraction and information retrieval systems, respectively.

2.6.1 Evaluating Information Extraction Systems

When evaluating an information extraction system, the systems' output is usually compared to some gold standard information, which is assumed to be correct and used as ground truth during the evaluation process. In the context of named entity recognition and normalization systems, this ground truth are usually annotations of human linguists (Nadeau and Sekine, 2007). In the following, we present the so-called confusion matrix classifying the decisions of an information extraction system into different groups of correct and incorrect decisions. Then, we present the frequently used measures of precision, recall, and f-score as well as the accuracy measure.

[16] While, in this section, we just briefly described UIMA's basics, which are required for understanding the later described tools and document processing pipelines, we refer for further information about UIMA to Ferrucci and Lally (2004a,b) and `http://uima.apache.org/` [last accessed April 8, 2014].

	gold standard (ground truth)	
system prediction	positive	negative
positive	TP	FP
negative	FN	TN

Table 2.3: General confusion matrix.

Confusion Matrix

Information extraction tasks can often be considered as specific sequential tagging and classification tasks (Weiss et al., 2005: p.132) and the confusion matrix (also called contingency table or contingency matrix) can be used to describe a system's errors compared to a gold standard (Manning and Schütze, 2003: p.268). In the following, we assume that in a gold standard, single instances, e.g., tokens, are either manually annotated as being a specific entity or as not being an entity. The system that is to be evaluated performs this task automatically, i.e., extracts entities of that type. All decisions of the information extraction system then can be grouped with the confusion matrix into one of the following four classes of a binary classification (Manning and Schütze, 2003: p.268):

- *true positive:* An instance, which is annotated by the system, is also annotated in the gold standard.
- *true negative:* An instance, which is not annotated by the system, is also not annotated in the gold standard.
- *false positive:* An instance, which is annotated by the system, is not annotated in the gold standard.
- *false negative:* An instance, which is not annotated by the system, is annotated in the gold standard.

In Table 2.3, the confusion matrix is depicted. Based on the four categories, the widely used evaluation measures of precision and recall can be calculated.

Precision, Recall, and F-score

Precision p (Equation 2.1; see, e.g., Manning and Schütze, 2003: p.268) is defined as ratio of instances correctly marked as positive by the system (TP) to all instances marked as positive by the system (TP+FP), with $0 \leq p \leq 1$. If all instances marked as positive by the system are correct, then precision equals 1. In contrast, if all instances marked as positive by the system are incorrectly marked, then precision equals 0.

$$\text{precision } p = \frac{\text{true positives (TP)}}{\text{true positives (TP)} + \text{false positives (FP)}} \tag{2.1}$$

Recall r (Equation 2.2; see, e.g., Manning and Schütze, 2003: p.269) is defined as ratio of instances correctly marked as positive by the system (TP) to all instances that should be marked as positive, i.e., to all instances marked as positive in the gold standard. As for precision, the range of recall is between 0 and 1 ($0 \leq r \leq 1$). Recall equals 0 if none of the instances that should be marked as positive are marked as positive by the system while recall equals 1 if all instances that should be marked as positive are also marked as positive by the system.

$$\text{recall } r = \frac{\text{true positives (TP)}}{\text{true positives (TP)} + \text{false negatives (FN)}} \tag{2.2}$$

Obviously, there is a trade-off between precision and recall. Marking all instances as positive results in a recall of 1 while marking only a single instance correctly as positive results in a precision of 1. Depending on the ratio of positive and negative instances in the gold standard, the other measures (precision and recall, respectively) would be rather low if these strategies were applied. Once a system already reaches a specific level for precision and recall, an increase of one of the measures usually involves a decrease of the other measure. Thus, the goal is often to find a good trade-off between precision and recall. To determine what "good" is, the f_β-score (also called f_β-measure) can be calculated (Equation 2.3; see, e.g., Manning et al., 2008: p.156). The f_β-score measures the weighted harmonic mean of precision and recall.

$$f_\beta\text{-score } f_\beta = \frac{(1+\beta^2) \times \text{precision (p)} \times \text{recall (r)}}{\beta^2 \times \text{precision (p)} + \text{recall (r)}} \tag{2.3}$$

Depending on the choice of β, precision and recall can be weighted differently. Frequently used values for β are 0.5, 1, and 2. The $f_{0.5}$-score weights the precision twice while the f_2-score weights the recall twice. Most frequently used is the f_1-score to calculate the balanced harmonic mean (Equation 2.4). Thus, it is often also referred to as f-score or f-measure.

$$f_1\text{-score } f_1 = \frac{2 \times \text{precision (p)} \times \text{recall (r)}}{\text{precision (p)} + \text{recall (r)}} \tag{2.4}$$

Note that in the context of named entity recognition and normalization (cf. Section 2.2), the measures precision, recall, and f-score can be calculated for the extraction subtask or for the full task of extraction and normalization. In the first case, an entity is considered as true positive (TP) if it is extracted by the system and marked in the gold standard. In the latter case, an entity is only considered as true positive if it is extracted by the system and marked in the gold standard, and, if it is additionally normalized correctly.

Calculating the measures of precision, recall, and f-score to evaluate the extraction as well as the extraction and normalization quality of information extraction systems allows to interpret their evaluation results and to compare different systems in a meaningful way.

Accuracy

An additional way to evaluate the quality of an information extraction system is to calculate the accuracy (Equation 2.5; see, e.g., Manning and Schütze, 2003: p.269). The difference between precision and accuracy is that precision deals only with a system's decisions about those instances marked as positive in the gold standard, i.e., only true positives (TP) and false positives (FP) are considered. In contrast, accuracy calculates the correctness of all decisions independent of whether instances are marked as positive or negative in the gold standard. Thus, accuracy is calculated as the ratio of correct decisions to all decisions.

$$\text{accuracy} = \frac{\text{correct decisions}}{\text{all decisions}} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{2.5}$$

Note that for some applications, accuracy is the typically used evaluation measure, e.g., for tokenization and sentence splitting applications (Tomanek et al., 2007). However, for information extraction systems, precision, recall, and f-score are typically used because in many situations the class of true negatives "is huge and dwarfs all the other numbers" (Manning and Schütze, 2003: p.269) resulting in high accuracy values independent of how well the class of positives (e.g., entities) is handled by the system. Nevertheless, accuracy is sometimes also reported. For instance, in the context of named entity extraction and normalization, the subtask of normalization is sometimes evaluated based on the accuracy measure. Note, however, that the set of all decisions then does not contain decisions about all entities but only about those extracted by the system (Equation 2.6). Thus, the class of false negatives only contains extracted entities not correctly normalized by the system.

$$\text{accuracy} = \frac{\text{correctly normalized entities}}{\text{correctly extracted entities}} \tag{2.6}$$

The normalization quality of two systems with different recall values in the extraction task should not be directly compared based on the accuracy value without considering the recall of the extraction. A system can achieve a higher accuracy score than another system although the latter normalizes more entities correctly. Assuming system A correctly extracts only one entity and also normalizes this entity correctly. Furthermore, assuming system B correctly extracts all entities in a data set and normalizes all entities except of one correctly, then, $1 = \text{accuracy(system A)} > \text{accuracy(system B)}$.

Due to this behavior, we prefer to evaluate both subtasks of named entity extraction and normalization using the measures precision, recall, and f-score. However, sometimes, accuracy is calculated for the normalization subtask as in the temporal tagging task of the TempEval-2 competition (Verhagen et al., 2010) for evaluating the normalization performance of so-called temporal taggers (cf. Section 3.6.2).

2.6.2 Evaluating Information Retrieval Systems

There are several ways to evaluate an information retrieval system. For instance, one can distinguish between evaluations in batch mode – a single query results in a particular system answer – and those in interactive sessions (Baeza-Yates and Ribeiro-Neto, 1999: p.74). While the latter ones require the analysis of user behavior in a series of interactive steps with the system, evaluations in batch mode can be considered as laboratory experiments and are repeatable (Baeza-Yates and Ribeiro-Neto, 1999: p.74). Furthermore, there are evaluation measures for unranked and ranked retrieval scenarios (Manning et al., 2008: p.155). In unranked retrieval scenarios, the retrieved documents are considered as an unordered set of documents, while in ranked scenarios the ordering provided by the system comes into play.

As in almost all academic research works, only batch-style evaluations are considered in this work. In the following, evaluation methods for those experiments will be surveyed. After briefly explaining measures for unranked retrieval, we present some further measures for ranked retrieval.

Unranked Retrieval Evaluation Measures: Precision, Recall, and F-score

Similar as information extraction, unranked information retrieval can be considered as binary classification task (Manning et al., 2008: p.152). Given a collection of documents and an information need expressed by a query, documents can be classified as being relevant or non-relevant, and the system either retrieves a document or it does not retrieve a document for a query. Thus, the decisions of a system can again

	document collection (ground truth)	
system prediction	relevant	non-relevant
retrieved	TP	FP
not retrieved	FN	TN

Table 2.4: Confusion matrix in the context of information retrieval.

be classified as true positives, false positives, true negatives, and false negatives. As shown in Table 2.4, instead of "positive" and "negative", one uses the terms of "relevant", "non-relevant", "retrieved", and "not retrieved" when evaluating information retrieval systems (Manning et al., 2008: p.155).

Based on the categories in the confusion matrix, one can easily calculate precision and recall as already detailed in Equation 2.1 and Equation 2.2, respectively. However, in the context of information retrieval, it is more intuitive to formulate precision as ratio of retrieved relevant items to all retrieved items, and recall as the ratio of retrieved relevant items to all relevant items as done in Equation 2.7 and Equation 2.8, respectively (Manning et al., 2008: p.155).

$$\text{precision } p = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{\#(\text{relevant items retrieved})}{\#(\text{retrieved items})} \tag{2.7}$$

$$\text{recall } r = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\#(\text{relevant items retrieved})}{\#(\text{relevant items})} \tag{2.8}$$

For a meaningful combination of precision and recall, the f-score can again be calculated (Equation 2.4, page 29). However, in the presented form, precision, recall, and f-score can only be used to evaluate systems in unranked information retrieval scenarios. To evaluate systems in ranked retrieval scenarios, they have to be adapted or other evaluation measures have to be used, as will be detailed next.

Ranked Retrieval Evaluation Measures: Precision at k, Average Precision at k, nDCG at k

When evaluating a system in a ranked retrieval scenario, "appropriate sets of retrieved documents are naturally given by the top k retrieved documents" (Manning et al., 2008: p.158). Thus, a simple evaluation measure in ranked retrieval scenarios is *precision at* k (Equation 2.9).

$$\text{precision at k } p@k = \frac{\#(\text{retrieved relevant items ranked } \leq k)}{\#(\text{retrieved items ranked } \leq k)} \tag{2.9}$$

A shortcoming of precision at k is that it does not consider the ranking within the set of the top k documents. For instance, two systems A and B could have a precision at k of 0.5 if both systems retrieve $k/2$ relevant documents. However, system A might have ranked the documents on rank 1 to $k/2$ while system B might have ranked them $k/2 + 1$ to k. Clearly, one would prefer system A over system B since its ranking is much better.

An evaluation measure taking into account the ranking within the set of the top k documents is *average precision* since it "aggregates many precision numbers into one evaluation figure" (Manning and Schütze,

2003: p.535). Usually, "precision at relevant documents that are not in the returned set is assumed to be zero" (Manning and Schütze, 2003: p.536).

However, instead of considering all relevant documents for a query – which requires that each document in a document collection is annotated as being relevant or non-relevant – average precision can also be used with a fixed cut-off level, i.e, as *average precision at k*. As formulated in Equation 2.10 (following Yue et al., 2007), it is calculated as the sum of precision scores (p@j) at each rank j of a relevant document retrieved by the system ($p_j = 1$) for ranks smaller or equal k. This sum is then averaged by either the number of relevant documents for the query (rel) or k (if there are more than k relevant documents).

$$\text{average precision at k } ap@k = \frac{1}{min(rel, k)} \sum_{j:p_j=1}^{k} p@j \tag{2.10}$$

In some evaluation scenarios, documents are not only classified binary as relevant or non-relevant but are graded according to how relevant they are. While average precision at k takes into account the ranking of documents, it only considers whether a document is relevant and does not distinguish if the relevance of documents is graded. Intuitively, highly relevant documents are more valuable than marginally relevant documents (Järvelin and Kekäläinen, 2002) and should be ranked higher than lower graded ones. Thus, an evaluation measure suitable for graded relevance judgments should penalize if highly graded documents are ranked lower.

An evaluation measure taking non-binary relevance judgments into account is (normalized) discounted cumulative gain as depicted in Equation 2.11 (Järvelin and Kekäläinen, 2002). By dividing the relevance judgments of all retrieved documents with a rank $i > 1$ by $log_2(i)$, the maximal score that can be achieved is lower, the higher the rank i. Thus, an optimal ranking has to order the documents by its relevance scores.[17]

$$\text{discounted cumulative gain at k } DCG@k = rel_1 + \sum_{i=2}^{k} \frac{rel_i}{log_2(i)} \tag{2.11}$$

An alternative calculation of discounted cumulative gain at k is formulated in Equation 2.12 (see, e.g., Manning et al., 2008: p.163). Note that the two equations 2.11 and 2.12 do not result in identical scores. However, they share the behavior of penalizing highly relevant documents being ranked lower.

$$\text{discounted cumulative gain at k (alternative) } DCG@k = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{log_2(i+1)} \tag{2.12}$$

Independent of whether using Equation 2.11 or Equation 2.12, the DCG@k measure can be normalized so that a perfect ranking results in a score of 1. For this, DCG@k is divided by the ideal discounted cumulative gain at position k (IDCG@k), i.e., by the score for a perfect ranking.

$$\text{normalized discounted cumulative gain at k } nDCG@k = \frac{DCG@k}{IDCG@k} \tag{2.13}$$

[17] Note that it is neither necessary to use the logarithm base 2 nor to use a logarithmic discount at all. However, using the base 2 logarithm results in a smooth reduction (Järvelin and Kekäläinen, 2002).

While there are several further measures to evaluate information retrieval systems (see, e.g., Baeza-Yates and Ribeiro-Neto, 1999; Manning et al., 2008), the presented measures of precision at k, average precision at k, and normalized discounted cumulative gain will be used in this work (e.g., in Chapter 5).

Note that all three measures have been formulated to evaluate single queries. Of course, they can be used for sets of queries by summing over all scores for the queries and dividing the sum by the number of queries (Manning et al., 2008: p.160). In the case of average precision, the measure is then called mean average precision. Note that "each information need [is weighted] equally in the final reported number, even if many documents are relevant to some queries whereas very few are relevant to other queries" (Manning et al., 2008: p.161).

2.7 Summary of the Chapter

In this chapter, we placed the thesis into its research context of natural language processing and text mining, and in particular of information extraction and information retrieval. In addition, we introduced basic concepts such as named entity recognition and its subtasks of recognition, classification, and normalization. We pointed out the importance of the normalization subtask which makes it possible to not only extract named entities as a specific type of entity (e.g., a person) but which also allows to link entity mentions to respective entities (e.g., a specific person).

Furthermore, the concepts of temporal and geographic information have been introduced, their key characteristics have been explained and compared, and it was described how temporal and geographic information – in particular temporal and geographic expressions – occur in textual documents. Finally, we briefly described the basics of the UIMA framework and presented several evaluation measures which we do not only refer to when evaluating our own approaches in this work but also when discussing related approaches.

In the next chapter, we will address the information extraction task of temporal tagging by surveying the research area and presenting our contributions to the research field.

3 Cross-domain Temporal Tagging

In this chapter, the task of temporal tagging is addressed with a special focus on multilingual and cross-domain temporal tagging. In Section 3.1, we demonstrate the importance of temporal taggers and give a brief description of the research field of temporal information extraction, of which temporal tagging is a subtask. The state-of-the-art of temporal tagging is described in Section 3.2 by presenting annotation standards, research competitions, annotated corpora, as well as state-of-the-art approaches to temporal tagging and existing temporal taggers. This section will be closed with a discussion of open issues.

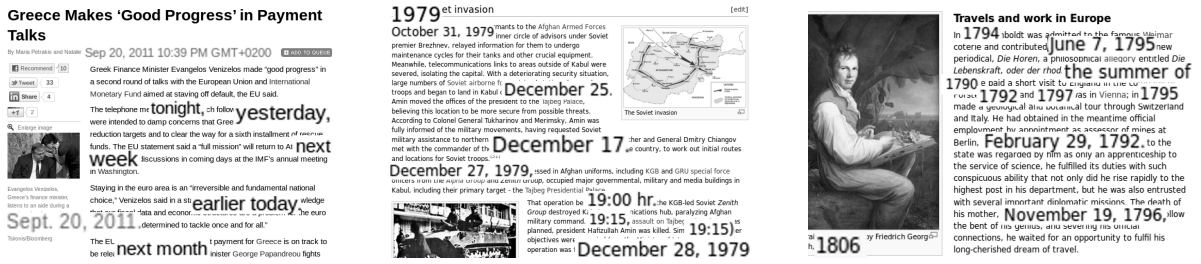
Motivated by these open issues in state-of-the-art temporal tagging, we then outline differences and challenges of temporal tagging documents from different domains (genres) in Section 3.3 and multilingual temporal tagging in Section 3.4. These differences and challenges in domain-sensitive and multilingual temporal tagging and since they have rarely been addressed in previous work are also the main reason why we developed the multilingual, cross-domain temporal tagger HeidelTime, which is detailed in Section 3.5. After an extensive evaluation of HeidelTime in Section 3.6, we close the chapter by discussing possible future work related to HeidelTime (Section 3.7) and summarizing the chapter of temporal tagging (Section 3.8).

3.1 Introduction and Motivation

The task of *temporal tagging* can be defined as the extraction and normalization of temporal expressions from text documents according to some annotation guidelines. In general, temporal tagging is thus a specific type of named entity recognition and normalization (cf. Section 2.2).

Since temporal information is prevalent in many kinds of documents – as it is exemplarily shown in Figure 3.1 for three types of documents – the extraction and normalization of temporal expressions from documents are important preprocessing steps for many natural language processing and understanding tasks. For example, in information retrieval, temporal information can be used, among others, for temporal clustering of documents along timelines and querying a document collection using temporal constraints – an issue also addressed later in this thesis (Chapter 5). While Alonso et al. (2007) gave an overview of the value of temporal information in information retrieval, we described a wide range of research trends in temporal information retrieval together to re-emphasizes this importance (Alonso et al., 2011).

Information Retrieval is usually not the research area for which rich natural language understanding is necessary. Thus, in research areas requiring rich natural language understanding, such as information extraction, document summarization, machine translation, and question answering, temporal information is often utilized a fortiori. For example, the ultimate goal of temporal information extraction can be summarized as "[t]he automatic identification of all temporal referring expressions (timexes), events, and temporal relations within a text" (UzZaman et al., 2013). Thus, a prerequisite of the final task of temporal annotation, i.e., the identification of temporal relations between events, and between events and temporal expressions, is to extract and normalize temporal expressions.



(a) News document. (b) History document. (c) Biography document.

Figure 3.1: Examples of three types of documents in which temporal information occurs frequently.
Sources: (a): `http://www.bloomberg.com/`.
(b): `http://en.wikipedia.org/wiki/Soviet_war_in_Afghanistan`.
(c): `http://en.wikipedia.org/wiki/Alexander_von_Humboldt`.

Independently of their specific goal, all applications using temporal information mentioned in text documents rely on high quality temporal taggers. Due to its importance for many tasks, temporal tagging has become an active research field over the past few years. This resulted in the development of standards for temporal annotation, the creation of annotated corpora, as well as several competitions, which were organized to evaluate temporal taggers. These topics are surveyed in the following section together with an overview of state-of-the-art approaches to temporal tagging and existing temporal taggers.

3.2 State-of-the-Art in Temporal Tagging

As introduced in Section 2.3.2, there are different types of temporal expressions such as date, time, duration, and set expressions. In addition, temporal expressions can carry their meaning explicitly, implicitly, or relative to some context information. Thus, when addressing the task of temporal tagging, it is necessary that it is well defined (i) what types of temporal expressions are "markable" (Ferro et al., 2005b) and should thus be annotated, (ii) to what extent expressions should be annotated, and (iii) how the semantics of the expressions can be captured by using normalization attributes requiring some values in standard format.

To address these questions, annotation standards and guidelines have been developed over the past few years (Section 3.2.1), research competitions have been organized to evaluate temporal-aware systems with respect to some annotation guidelines (Section 3.2.2), and corpora have been annotated according to them (Section 3.2.3).

3.2.1 Annotation Standards

Currently, there are two annotation standards used for annotating temporal expressions in documents: The TIDES TIMEX2 standard (Ferro et al., 2001, 2005b) and TimeML (Pustejovsky et al., 2003a, 2005), a specification language for temporal annotation containing TIMEX3 tags for temporal expressions. Both standards present guidelines for the annotation of temporal expressions, including how to determine the extents of expressions and their normalizations. In both cases, the normalization is defined according to the ISO 8601 standard for temporal information with some extensions. Since all widely used annotated corpora (cf. Section 3.2.3) as well as all state-of-the-art systems (cf. Section 3.2.5) are based on either one of the two standards, we describe the details of both of them in the following.

TIDES TIMEX2 Annotation Standard

While there had been several TIMEX definitions reaching from extent-only coverage (see, e.g., Chinchor, 1997), up to inclusion of some normalization information (see, e.g., Mani and Wilson, 2000a; Setzer and Gaizauskas, 2000), the TIDES TIMEX2 definitions were the first annotation guidelines being well-defined with sufficient detail to became broadly accepted as a standard. The annotation guidelines are based on the principles that temporal expressions should be tagged "if a human can determine a value for [it]", and that the value "must be based on evidence internal to the document" (Ferro et al., 2001). Covering extent and normalization information, both questions *What is a temporal expression?* as well as *What is the meaning of a temporal expression?* are addressed. For the normalization, TIMEX2 tags may contain the following attributes (Ferro et al., 2005b):

- *VAL:* a normalized form of the date/time [or duration/set]
- *MOD:* captures temporal modifiers
- *ANCHOR_VAL:* a normalized form of an anchoring date/time
- *ANCHOR_DIR:* the relative direction between VAL and ANCHOR_VAL
- *SET:* identifies expressions denoting sets of times

TimeML Annotation Standard

TimeML is based on the TIDES standard and was developed to capture further types of temporal information in documents: events, temporal relations between events and temporal expressions, as well as temporal relations between two events. Thus, in contrast to the single tag approach of the TIDES annotation standard, TimeML contains tags for annotating events, temporal links, and temporal signals in addition to the TIMEX3 tag for temporal expressions (Pustejovsky et al., 2003a, 2005). Due to this extension of annotating temporal information, there are significant differences between TIMEX2 and TIMEX3. These affect the attribute structure as well as the exact use. For example, events can be part of temporal expressions in TIMEX2 (<TIMEX2>two days after the revolution</TIMEX2>), while they are not part of temporal expressions following TimeML (<TIMEX3>two days</TIMEX3> after the revolution).

More generally, specific types of pre- and post-modifications of temporal expressions are part of TIMEX2 tags while in TimeML, they are outside TIMEX3 tags (Mazur, 2012). Such constructs are handled using the newly introduced tags for annotating relations between temporal expressions and events. In addition, TIMEX3 tags cannot be nested. However, TIMEX3 tags with no extent are introduced, e.g., to deal with unspecified time points, which are needed to anchor durations. Note, however, that despite the fact that such abstract tags, i.e., annotations without any extent, are described in the TimeML annotation guidelines (Mazur, 2012), they are practically never used – neither in annotated corpora nor by TIMEX3-compliant temporal taggers. The most important attributes of TIMEX3 tags are[1]:

- *type:* defines whether the expression is of type date, time, duration, or set
- *value:* a normalized form of the expression

[1]The details of the attributes are described in the TimeML annotation guidelines including further attributes, e.g., to capture the function of a temporal expression in a document. For details, see `http://www.timeml.org/` [last accessed April 8, 2014].

- *mod:* captures temporal modifiers
- *quant* and *freq:* specifies the quantity and frequency of set expressions
- *beginpoint* and *endpoint:* anchor begin and end of a duration

While the attribute *type* – with possible values "date", "time", "duration", and "set" – is newly introduced in TIMEX3, the attributes *value* and *mod* are similar to the VAL and MOD attributes in TIMEX2. These two attributes already capture a large part of the information of temporal expressions, and for many expressions, the *value* attribute is the only attribute that is needed for normalization. This is also the reason why in several evaluations of temporal taggers, the *value* attribute is the focus of interest (see, e.g., UzZaman et al., 2013). Although the different attributes and definitions of extents between TIMEX2 and TIMEX3 are significant, the annotations for many temporal expressions are very similar, and an automated conversion works reasonably well (see, Saquete, 2010; Saquete and Pustejovsky, 2011).

Using TimeML and TIDES TIMEX2 annotation standards, several research competitions have been organized, and several corpora have been manually annotated to be used as benchmarks. In the following sections, we survey temporal tagging research competitions and present an overview of existing annotated corpora and also approaches to translate TIMEX2 annotations of annotated corpora into TIMEX3.

3.2.2 Research Competitions

The first research competitions addressing the extraction of temporal expressions were the MUC (Message Understanding Conference) named entity recognition tasks in 1995 (Grishman and Sundheim, 1995) and 1997 (Chinchor, 1997). In addition to named entities of the types person, organization, and location, numeric and temporal expressions had to be detected by the participants' systems. For the annotation of temporal expressions, TIMEX tags were introduced, which later became the basis for the development of the TIDES TIMEX2 and TimeML TIMEX3 standards.

It took several more years until the first research competition was organized addressing not only the extraction of temporal expressions but also their normalization. In the ACE (Automatic Content Evaluation) time expression and normalization (TERN) contest in 2004, as well as in the two follow-up contests in 2005 and 2007, temporal expressions had to be detected and normalized according to TIDES TIMEX2 annotation standard.

With the organization of the first TempEval competition (Verhagen et al., 2007, 2009) as part of the SemEval series, temporal relation extraction became the main goal in the research community and the temporal markup language TimeML was used as annotation standard. Given documents annotated with temporal expressions and events according to the TimeML standard, the task of the participants was to develop systems to automatically determine temporal relations between events and the document creation time, between temporal expressions and events, and between two events in consecutive sentences. In TempEval-2 (Verhagen et al., 2010) and TempEval-3 (UzZaman et al., 2013), temporal tagging and event extraction and normalization have been added as subtasks to offer research competitions for the full task of temporal information extraction. The goal of the temporal tagging subtasks was to extract and normalize temporal expressions following TimeML's TIMEX3 definition. Since we participated in TempEval-2 and TempEval-3 with our temporal tagger HeidelTime (cf. Section 3.5), more details about these contests will be given in Section 3.6 when we present HeidelTime's evaluation results.

corpus name	text types	standard	documents	annotations[1]
ACE TERN 2004 training	news	TIMEX2	862	8,938
ACE TERN 2004 evaluation	news	TIMEX2	192	
ACE Multilingual 2005 training	news, conversation, discussion, Weblog	TIMEX2	599	5,469
TIDES Parallel Temporal Corpus	dialogs	TIMEX2	95	3,481[2]
Timebank v1.2	news	TIMEX3	183	1,414
Timebank (TempEval-3 version)	news	TIMEX3	183	1,426
AQUAINT	news	TIMEX3	73	605
AQUAINT (TempEval-3 version)	news	TIMEX3	73	652
TempEval-2 (temporal tagging)	news	TIMEX3	9	81*
TempEval-2 (combined test sets)	news	TIMEX3	20	156*
TempEval-3 platinum evaluation	news	TIMEX3	20	158
TimenEval	news, others	TIMEX3	9	214
WikiWars	Wikipedia	TIMEX2	22	2,681

Table 3.1: Statistics on publicly available English corpora containing TIMEX2 or TIMEX3 annotations.
[1]Document creation times (DCT) are included except for corpora marked with *.
[2]According to Derczynski et al. (2012); Mazur (2012) report 3,541 temporal expressions.

In the context of the described research competitions, several corpora have been developed as training and evaluation data sets. In the next section, we present these corpora, give an overview of further annotated corpora and also approaches to translate TIMEX2 into TIMEX3 annotations.

3.2.3 Annotated Corpora

In this section, we focus on the description of English corpora annotated with temporal expressions according to the TIDES TIMEX2 or TimeML annotation standard. In Section 3.4, we will present more information about non-English corpora when discussing multilingual temporal tagging. Table 3.1 shows some statistics of the corpora, of which several are also used to evaluate HeidelTime.

The ACE (TERN) Corpora

The ACE (TERN) corpora were developed in the context of the ACE time expression and normalization contests. Although all training and evaluation sets of ACE 2004, 2005, and 2007 were annotated using TIMEX2 tags, different versions of the annotation guidelines were used (Mazur, 2012). The changes, however, are not significant. So far, only the 2004 training, the 2004 evaluation, and the 2005 training corpora have been released by the Linguistic Data Consortium.[2] In contrast, no new training corpus was developed in the context of the ACE 2007 contest, and the ACE 2005 and 2007 evaluation corpora are not distributed so far. The three publicly available ACE corpora are created in a similar format and there are official evaluation scripts, which can be used after slight modifications for evaluating temporal taggers.

[2]ACE TERN 2004 training corpus, ACE TERN 2004 evaluation corpus, and ACE Multilingual 2005 training corpus are released by the Linguistic Data Consortium (LDC), catalog numbers LDC2005T07, LDC2010T18, and LDC2006T06, respectively; `http://www.ldc.upenn.edu/` [last accessed April 8, 2014].

The ACE TERN 2004 training corpus (Ferro et al., 2005a) contains 862 English documents from the news domain including 95 English translations of the Arabic Treebank and Chinese Treebank. The documents are annotated with TIMEX2 tags and contain 8,938 annotated temporal expressions. The ACE TERN 2004 evaluation corpus (Ferro et al., 2010) contains 192 news documents also annotated with TIMEX2 tags.[3]

The ACE Multilingual 2005 training corpus (Walker et al., 2006) consists of English, Arabic, and Chinese documents. Although the Arabic and Chinese documents are annotated with TIMEX2 tags, they only contain extent information for the temporal expressions and no normalization information is provided. In Section 3.4, we will present more information about non-English corpora annotated with temporal expressions in general, and we will also get back to the ACE Multilingual 2005 training corpus. In contrast to the Arabic and Chinese documents, the 599 English documents are annotated with both, extent and normalization information. The documents are from the news domain but also texts from conversations, discussions, and Weblogs are included. In total, they contain 5,469 TIMEX2 annotations.

The TIDES Parallel Temporal Corpus

Another corpus annotated according to the TIDES annotation guidelines is the TIDES Parallel Temporal Corpus. It contains transcriptions of 95 dialogs about arranging meetings, and the original Spanish conversations[4] have been translated into English. Although the corpus is rich in temporal information (3,481 TIMEX2 annotations), it misses valuable information such as when the dialogs took place. Thus, underspecified and relative temporal expressions cannot be fully normalized due to missing context information, and the corpus is rarely used to evaluate temporal taggers.

The TimeBank Corpus

The Timebank corpus was initially developed during the Time and Event Recognition for Question Answering Systems (TERQAS) workshop in 2002 as a reference corpus for TimeML (Pustejovsky et al., 2003b). Thus, TimeBank contains TIMEX3 tags for temporal expressions, and events and temporal relations are also annotated. The TimeBank 1.2 version released by the Linguistic Data Consortium[5] consists of 183 news documents with 1,414 TIMEX3 annotations. For the TempEval-2 and TempEval-3 competitions, TimeBank was provided as training corpus (Verhagen et al., 2010; UzZaman et al., 2013) and in the context of TempEval-3 a cleaned and improved version of TimeBank was released[6] now containing 1,426 TIMEX3 annotations.

The AQUAINT Corpus

Similar to the TimeBank corpus, the AQUAINT corpus[7] also contains news documents annotated according to the TimeML annotation standard. However, it "is not as mature as TimeBank 1.2 [...] [since the annotators] did not go through several rounds of annotation and annotation reviews" (Verhagen and Moszkowicz, 2008). The AQUAINT corpus was used as additional training corpus in the TempEval-3 competition and a cleaned version is available together with the latest TimeBank corpus. This version of the AQUAINT corpus contains 73 documents with 652 TIMEX3 annotations.

[3] The recently released version has hardly been used for evaluating temporal taggers yet. Since we do not have access to the corpus, we cannot report any further details such as the number of annotated temporal expressions.

[4] The Spanish dialogs are part of the Enthusiast corpus (Suhm et al., 1994).

[5] TimeBank is released by LDC, catalog number LDC2006T08; `http://www.ldc.upenn.edu/` [last accessed April 8, 2014].

[6] `http://www.cs.york.ac.uk/semeval-2013/task1/` [last accessed April 8, 2014].

[7] `http://timeml.org/site/timebank/timebank.html` [last accessed April 8, 2014].

The TempEval Corpora

In the context of TempEval-2 and TempEval-3 manually annotated corpora were created by the organizers. In addition to the corpora, official evaluation scripts are publicly available.[8]

For the TempEval-2 challenge (Verhagen et al., 2010), the TimeBank corpus was used as training data – although in a different format than the original corpus. For the evaluation, new data sets were manually annotated. The TempEval-2 evaluation corpus for the temporal tagging task consists of 9 documents containing 81 annotated temporal expressions. While this corpus was used to evaluate the temporal tagging performance, an additional data set has been annotated to evaluate the temporal relation task of TempEval-2 since here, TIMEX3 annotations were provided to the systems. Putting both sets together, the whole TempEval-2 evaluation set contains 20 documents and 156 TIMEX3 annotations. However, the test set of TempEval-2 is still rather small compared to other publicly available corpora (cf. Table 3.1).

In addition to the TimeBank and AQUAINT corpora, the organizers of TempEval-3 (UzZaman et al., 2013) provided a large silver standard as additional training corpus. However, since all annotations were created by automatically merging annotations of three systems (Llorens et al., 2012b), the quality of this corpus is not sufficient to use it as benchmark for evaluating temporal taggers in a meaningful way. Thus, we do not provide any further details about this corpus here. In contrast, the newly developed TempEval-3 platinum corpus[9] is of high quality (UzZaman et al., 2013). In total, the 20 documents contain 158 TIMEX3 annotations. However, as the test set of TempEval-2, the TempEval-3 platinum corpus contains rather few temporal expressions compared to other publicly available corpora (cf. Table 3.1).

The WikiWars Corpus

The WikiWars corpus (Mazur and Dale, 2010) consists of 22 documents with parts from Wikipedia articles about important wars in history. Thus, it has been the first corpus annotated with temporal expressions, which contains narrative text and not news or news-style documents – an important difference to all other corpora as will be pointed out in Section 3.3. In total, the 22 documents contain 2,681 temporal expressions annotated according to TIDES TIMEX2 annotation guidelines. WikiWars is publicly available[10] and formatted in the same style as the ACE corpora so that the same evaluation scripts can be used to determine a temporal tagger's extraction and normalization quality.

The TimenEval Corpus

The TimenEval corpus[11] consists of nine documents with 214 TIMEX3 annotations and was developed to evaluate the temporal expression normalization resource TIMEN (Llorens et al., 2012a), which will be described in Section 3.2.5. TimenEval contains "a significant amount of non-newswire material" (Llorens et al., 2012a) in addition to some news document. This mixture between news-style and non-news-style documents, however, is critical since processing documents from different domains benefits from domain-dependent normalization strategies as we will detail in Section 3.3 where we discuss the temporal-tagging-relevant characteristics of documents from different domains. Thus, and in addition due to several

[8] See `http://www.timeml.org/site/timebank/tempeval/tempeval2-data.zip` [last accessed April 8, 2014] and `http://www.cs.york.ac.uk/semeval-2013/task1/` [last accessed April 8, 2014] for evaluation tools.

[9] `http://aclweb.org/aclwiki/code/5/51/ADCR2013T001.tar.gz` [last accessed April 8, 2014].

[10] `http://www.timexportal.info/wikiwars/` [last accessed April 8, 2014].

[11] `http://code.google.com/p/timen/source/browse/#svn%2Ftrunk%2Feval_corpus%2Fmerged` [last accessed April 8, 2014].

annotation differences to other corpora,[12] in our opinion, this corpus should not be used directly without preprocessing for evaluating temporal taggers, and we do not consider this corpus in our evaluation.

Translation from TIMEX2 to TIMEX3

Due to the increasing popularity of TimeML and the large number of corpora annotated according to TIDES TIMEX2 annotation guidelines, there have been approaches to automatically translate TIMEX2 annotations to TimeML. Saquete and Pustejovsky (2011) started the development of the T2T3 transducer to convert TIMEX2 annotations to TIMEX3 (and additional TimeML tags if required). While they only performed a small evaluation using the TimeBank corpus and parts of the ACE TERN 2004 corpus, Derczynski et al. (2012) extended this work. They applied the new T2T3 transducer to the following corpora and made the TIMEX3-versions of these corpora publicly available:[13] ACE TERN 2004 training corpus, ACE Multilingual 2005 training corpus, TIDES Parallel Temporal Corpus, and WikiWars. However, when reporting HeidelTime's evaluation results on these corpora in Section 3.6, we will not use the TIMEX3 versions of the TIMEX2 corpora since the annotations are translated automatically and the manually corrected versions of the corpora are not yet available.

3.2.4 State-of-the-Art Approaches to Temporal Tagging

As described in Section 3.1, the task of temporal tagging can be split into two subtasks, the extraction and the normalization of temporal expressions. The extraction task is to correctly identify temporal expressions and their boundaries in a text document. It can thus be regarded as a typical classification problem of deciding whether or not a token is part of a temporal expression. For this, approaches range from rule-based to machine learning strategies to extract temporal expressions.

Rule-based Approaches for the Extraction Task

Rule-based approaches usually make use of at least some of the following features: pattern lists, regular expressions, part-of-speech information, positive or negative constraints, and cascaded organization of rules. Some existing rule-based temporal taggers and their strategies will be detailed in the next section.

Machine-learning Approaches for the Extraction Task

Machine-learning approaches typically rely on a variety of features, which are frequently divided into four groups as suggested by, e.g., Hacioglu et al. (2005) and Mazur (2012): lexical features (e.g., token, part-of-speech, character-based features, frequency), syntactical features (e.g., base phrase chunks), semantic features (e.g., semantic role labels), and external features (tags of temporal expressions identified by other taggers). While using additional temporal taggers, i.e., external features, is rather untypical, if it is applied, then only by machine learning approaches. In contrast, many of the other features exploited by machine learning approaches are often also used by rule-based approaches.

[12] For instance, one document (eng-WL-11-174646-13000523.tml) in the corpus is very long containing pieces of text from forum discussions without marking the single parts of the conversation. Despite that, a single document creation time (DCT) is provided for the full document. While this DCT is used to normalize relative and underspecified expressions at the beginning of the document, later expressions are not disambiguated according to the DCT. For example, there are several time expressions in the TimenEval corpus for which the value attribute only contains the time information without date information (e.g., <TIMEX3 value="T20:13:00" tid="t86" type="TIME">20:13:00</TIMEX3>). This mixture within a corpus, and in particular within a single document, is problematic when performing an evaluation on this corpus.

[13] `http://bitbucket.org/leondz/t2t3/` [last accessed April 8, 2014].

Based on the features, a range of different machine learning techniques can be trained using some training data, i.e., temporally annotated corpora. Some of these machine learning methods, which have been frequently applied for the task of temporal tagging, are: maximum entropy classifier, support vector machines, and conditional random fields. As for rule-based systems, we present some existing temporal taggers relying on machine-learning methods for the extraction task in the next section.

The Normalization Task

The goal of normalizing temporal expressions is to capture their temporal meaning. Thus, values in some standard format – usually following specific annotation guidelines (cf. Section 3.2.1) – for several attributes are assigned to each temporal expression. This is a more challenging and complex task than the extraction, and almost all temporal taggers address the normalization task in a rule-based way. In summary, existing temporal taggers use either a combination of machine learning and rule-based approaches or exclusively rule-based methods.

Document Types and Languages

While there has been a lot of research on temporal tagging in the last years, almost always news or news-style documents were addressed. This, however, is problematic due to different characteristics of text documents from different domains, as we will discuss in detail in Section 3.3. Similarly, most of the research on temporal tagging deals with English as the only language. In Section 3.4, related work on multilingual temporal tagging will be surveyed and language-dependent characteristics and challenges for temporal tagging several languages will be presented.

3.2.5 Existing Temporal Taggers

In this section, we survey existing temporal taggers with a focus on English temporal tagging. While we briefly comment the systems' performance, detailed evaluation results of the presented systems will be provided in Section 3.6 if the taggers were evaluated on publicly available corpora. There, we will directly compare the evaluation results of HeidelTime and the systems described in this section.

TempEx and GUTime

One of the first temporal taggers for extracting and normalizing temporal expressions is TempEx (Mani and Wilson, 2000a). It is a simple, rule-based system that uses TIMEX2 tags, although the normalization functionality is limited. Based on this temporal tagger, GUTime was developed as reference tool for TimeML using TIMEX3 tags.[14] For quite a long time, GUTime was one of the most widely used temporal taggers. It is part of the TARSQI toolkit consisting of components for the extraction of events, temporal expressions, and temporal relations (Verhagen and Pustejovsky, 2008, 2012). GUTime has been evaluated on the ACE TERN 2004 training data and achieves competitive results.

Chronos

Chronos (Negri and Marseglia, 2004) is a TIMEX2-compliant temporal tagger for English and Italian and was the best performing system at the ACE TERN 2004 competition (English) performing the full task of temporal tagging. For both, the extraction and the normalization, a rule-based approach is realized.

[14] `http://timeml.org/site/tarsqi/modules/gutime/index.html` [last accessed April 8, 2014].

The system architecture is split into a detection/bracketing component and a normalization component. For identifying the boundaries of expressions, Chronos applies a relatively large set of about 1,000 handcrafted rules. They detect all possible temporal expressions, determine their extent, and gather contextual information relevant for the normalization task (Negri and Marseglia, 2004). Then, the output of the basic rules is processed by a set of composition rules to solve conflicts such as overlapping expressions.

Based on the context information collected during the detection phase, the normalization component sets the values of all TIMEX2 attributes. For this, an anchor expression is identified for each – as called in our terminology (cf. Section 2.3.2) – relative or underspecified expression (i.e., non-explicit expression). Either the document creation time (e.g., for "today", "December", "next month") or the previously mentioned expression with a compatible granularity (e.g., for "the following month", "two years ago") is selected depending on the information about the expression gathered during the detection phase (Negri and Marseglia, 2004). Finally, TIMEX2's VAL attribute is set by either directly normalizing the expressions' context information or by performing some calculations between the anchor expression and the context information.

By achieving the best results at the ACE TERN 2004 challenge, Negri and Marseglia (2004) showed that their strategy works well for ACE TERN-style documents.

TERSEO

Another rule-based temporal tagger is part of the TERSEO system for event ordering (Saquete et al., 2006b). The temporal tagging process is split into an extraction and a normalization phase. First, a chart parser is applied with a language-specific grammar and temporal expressions are marked as absolute (explicit) expressions and others. In the second phase, the expressions are normalized according to TIDES TIMEX2 annotation standard either directly or – if necessary – after the normalization unit determined the reference date and performed the value calculation for the expression (Negri et al., 2006).

TERSEO was originally developed for Spanish and extended to process further languages such as Italian and English by automatic rule translation and (semi-)automatically developed parallel corpora (Negri et al., 2006; Puchol-Blasco et al., 2007). However, this process resulted in lower temporal tagging quality compared to a tagger tailored to the language of interest.

Some methods to extend TERSEO to other systems – and thus also more details about the system architecture – will be presented in Section 3.4.4, when surveying temporal taggers with the focus on multilingual approaches. In addition, Saquete (2010) participated in the TempEval-2 challenge using the TERSEO system with a TIMEX2 to TIMEX3 transducer so that we will also refer to TERSEO in Section 3.6.2, where we present the TempEval-2 evaluation results of all participating systems.

DANTE

Another temporal tagger separating the tasks of extraction and normalization is the DANTE tagger (Mazur and Dale, 2009). The extraction is done using a JAPE grammar (JAVA Annotation Pattern Engine), and the normalization is performed in a rule-based manner. DANTE annotates temporal expressions according to the TIMEX2 guidelines and was one of the systems that participated in the ACE 2007 competition where it achieved competitive results. The developers of DANTE also developed the WikiWars corpus and pointed out the challenges of normalizing temporal expressions when processing narratives instead

of news documents (Mazur and Dale, 2010). Running DANTE on WikiWars significantly decreased DANTE's temporal tagging performance.[15]

Systems at TempEval-2 and TempEval-3

In Section 3.6.2 and Section 3.6.3, we will present the evaluation results of all participating systems of the TempEval-2 and TempEval-3 competitions, respectively, and compare those results with HeidelTime's performance at these research challenges.

SUTime

Shortly before the TempEval-3 challenge, SUTime was developed by Chang and Manning (2012). It is a deterministic rule-based system and part of the Stanford CoreNLP package. Both, the extraction and the normalization are performed in a similar way as done by our temporal tagger HeidelTime. The developers performed an evaluation on the TempEval-2 test set and compared SUTime amongst others with HeidelTime. While SUTime partially outperformed HeidelTime in this evaluation (Chang and Manning, 2012), HeidelTime achieved better results in the TempEval-3 competition (UzZaman et al., 2013) for the full task of temporal tagging.

Tools for Normalization only

Recently, a couple of approaches have been developed to perform the task of normalizing temporal expressions independently of the extraction: TIMEN (Llorens et al., 2012a) was the first normalization only tool. It consists of a rule base that requires as input the expression itself but also the document creation time, information about the reference time, and the tense of the sentence. Since the detection of the correct reference time is one of the difficult parts during the normalization phase – and, in addition, domain-dependent as will be detailed in Section 3.3 – it is not enough to just perform the extraction subtask before applying TIMEN for the normalization.

Three further approaches to perform only the normalization task of temporal tagging are developed by Angeli et al. (2012), Angeli and Uszkoreit (2013), and Bethard (2013a). All three approaches run a parsing strategy to normalize temporal expressions. While Bethard (2013a) manually developed the parsing grammar, the approaches by Angeli et al. (2012) and Angeli and Uszkoreit (2013) present the first approaches to learn the task of normalization.

When comparing tools performing only the task of normalization with a temporal tagger performing the full task of temporal tagging, the following question regarding the evaluation setup has to be answered: *The normalization of which temporal expressions shall be compared?* Llorens et al. (2012a) compared HeidelTime with TIMEN by evaluating HeidelTime's normalizations and TIMEN's normalizations on all the expressions extracted by HeidelTime. While the results of TIMEN and HeidelTime were identical on standard corpora, TIMEN outperformed HeidelTime's normalization only on the TimenEval corpus, which can be explained by the characteristics of the TimenEval corpus (cf. Section 3.2.3). Ignoring the corpus-specific issues, we consider this evaluation setup as a fair comparison between a normalization tool and a full temporal tagger. However, one might argue that it prefers the full temporal tagger since the normalization tool might correctly normalize expressions that were not extracted and the normalization component of the temporal tagger is tailored to those expressions that are extracted.

[15] As HeidelTime, the new DANTE version distinguishes between news- and narrative-style documents (Mazur, 2012).

Angeli et al. (2012) performed two evaluations. While the second one is identical to the one performed by Angeli and Uszkoreit (2013) and described below, the first evaluation setup is as follows: They developed an extraction component based on Conditional Random Fields and ran their normalization approach on the output of that extraction tool. While they receive lower results than HeidelTime, they argue that their normalization tool could have performed better if the recall of the extraction component had been better. This is due to the fact that the normalization component could offer normalizations for all kinds of expressions not only for those that have been extracted successfully. Thus, despite the fact that HeidelTime outperformed the suggested system in this evaluation setup, and that we consider this evaluation setup a fair evaluation, it is not clear which tool performs a better normalization task.

In contrast, Angeli and Uszkoreit (2013) (and also Angeli et al. (2012) in their second evaluation setup) compared the results on gold extents of temporal expressions of manually annotated corpora, and thus forced temporal taggers such as HeidelTime to normalize all temporal expressions occurring in the gold standard. As will be detailed in Section 3.5, HeidelTime performs the extraction and normalization tasks together. If HeidelTime does not extract a specific expression, it does not provide any suggestions for a normalization. Thus, the following two issues occur:

(i) Some expressions are easy to normalize once it is determined if they should be extracted or not. For instance, expressions such as "future" (val="FUTURE_REF"), "previously" ("PAST_REF"), and "currently" ("FUTURE_REF") are sometimes annotated as temporal expressions in the gold standard corpora with the shown values but sometimes not. Despite the fact that such expressions are rather less important for tasks relying on temporal tagging (due to their imprecise normalized values), in the extraction task, it is a trade-off between precision and recall as will be detailed in the error analysis (Section 3.6.8) so that HeidelTime does not extract all of these expressions. Thus, while the normalization is trivial, HeidelTime is penalized for the decision of not extracting those expressions.

(ii) A similar issue is that HeidelTime may correctly normalize some expressions although HeidelTime extracts them only partially. Due to forcing HeidelTime to normalize gold extents, such correct normalizations are not considered as being correct in this evaluation setup.

Finally, Bethard (2013a) performs the normalization in the following way: HeidelTime performed the full task of temporal tagging, and all correct normalizations were considered – independently of whether expressions were extracted partially or completely. Bethard (2013a)'s system was given the correct extents and reference times from the gold standard. While we prefer this evaluation setup over the setup performed by Angeli and Uszkoreit (2013), the first issue, i.e., that simple expressions that are not extracted are not normalized by HeidelTime, remains.

In Section 3.6, we perform an extensive evaluation of HeidelTime. However, since all normalization tools require information about reference times – not provided by extraction-only tools – and since it is not clear how to perform a fair comparison between full temporal taggers and normalization-only tools, we do not compare HeidelTime's performance with the above approaches for normalization only.

Summary

In this section, we presented several existing temporal taggers, and, in addition, recent approaches performing only the task of normalizing temporal expressions independent of how they have been extracted. Despite the wide range of approaches, there are several open issues in the research area of temporal tagging as will be explained next.

3.2.6 Summary and Open Issues

The main findings of the previous two sections are summarized as follows:

- Due to its importance for many natural language processing and understanding tasks, there is a lot of research on temporal annotation in general, and on tagging in particular. Approaches for the extraction of temporal expressions range from hand-crafted rules to machine learning or also hybrid strategies, but the normalization is usually performed using rule-based approaches.
- Most of the research on temporal tagging addresses temporal tagging English documents. Although there are some promising approaches to address other languages than English, porting a temporal tagger from one language to another often resulted in lower temporal tagging quality. In addition, there is a lack of available language resources which help to bring forward this research. Thus, there are no publicly available temporal taggers for processing many languages reaching high quality for both tasks, the extraction and the normalization of temporal expressions.
- Except some very recent work, all the research on temporal tagging during the last years deals with news or news-style documents as domain of interest. This is problematic since switching the domain results in other challenges. Thus, running a temporal tagger on documents of another domain significantly decreases the temporal tagging quality as Mazur and Dale (2010) showed when processing Wikipedia documents with a temporal tagger developed for the news domain.

In the following sections, we will address the topics of *temporal tagging documents of different domains* (Section 3.3) and *multilingual temporal tagging* (Section 3.4). Then, in Section 3.5, we present our approach to address these open issues, HeidelTime, a system for multilingual, cross-domain temporal tagging.

3.3 Temporal Tagging Documents of Different Domains

In this section, we address the issue of temporal tagging documents of different domains. After concisely defining the concept of a domain (Section 3.3.1), we present domain-dependent characteristics, which are crucial for the task of temporal tagging (Section 3.3.2 to Section 3.3.5).

In Section 3.3.6, we describe our development of temporally annotated corpora containing documents of domains that have not been addressed so far. Based on these and publicly available corpora, we perform a comparative corpus analysis in Section 3.3.7 to study the differences between domains and point out domain-dependent challenges. Finally, we suggest strategies to address these challenges in Section 3.3.8.

3.3.1 The Concept of a Domain

When applying a temporal tagger to different kinds of text documents, the quality of both, the extraction and the normalization of temporal expressions differs significantly. For example, Mazur and Dale (2010) reported a serious quality drop when using their temporal tagger DANTE to process Wikipedia documents instead of news documents, for which the tagger was originally developed.

Motivated by these observations, we performed a study on temporal tagging documents of several sources: news documents, Wikipedia documents, short messages, and scientific abstracts (Strötgen and Gertz, 2012b). As will be described in the following, we showed that depending on the type of documents that are to be processed, different challenges occur and different temporal tagging strategies help to

address them. For this, we define that document types with the same characteristics relevant for the task of temporal tagging are from the same *domain*.

Definition 3.1. *(Domain)*
A *domain* in the context of temporal tagging is defined as a group of documents having the same characteristics relevant for the task of temporal tagging.

There are manifold terms that can be used to name the concept of "a group of documents having the same characteristics relevant for the task of temporal tagging", such as genre, registers, text types, domains, or styles (see, e.g., Lee, 2001). We chose the term "domain" mainly for the following reasons: (i) The term "genre" is probably most frequently used to classify text documents. However, it is usually assumed that the characteristics of documents that make them belong to the same genre are non-linguistic features (Biber, 1988; Lee, 2001) – a fact that is not valid in our case. Thus, we do not use "genre" to avoid any misinterpretations. (ii) The combination of two meanings of a *domain* according to the Merriam Webster dictionary[16] fits exactly our context: "a sphere of knowledge, influence, or activity" and "a region distinctively marked by some physical feature". On the one hand, a temporal tagger can be aware of different domains, i.e., spheres of knowledge, and on the other hand, the text documents of every domain can be distinctively marked by some (linguistic) features as will be explained in detail in the following.

In the next sections, we define four domains for temporal tagging: news-style, narrative-style, colloquial-style, and autonomic-style documents. While the different challenges for temporal tagging of news-style and narrative-style documents have initially been studied by Mazur and Dale (2010), we analyze two further domains, namely colloquial-style and autonomic-style documents (documents from scientific literature, for instance). In both domains, temporal information plays a crucial role, e.g., in SMS messages for communicating about upcoming events or meetings, and in biomedical documents – as representative of scientific texts – for describing chronological procedures such as clinical trials.

Since there were no temporally annotated corpora with documents of these domains available so far, we created two new corpora and manually annotated the temporal expressions occurring in the documents.

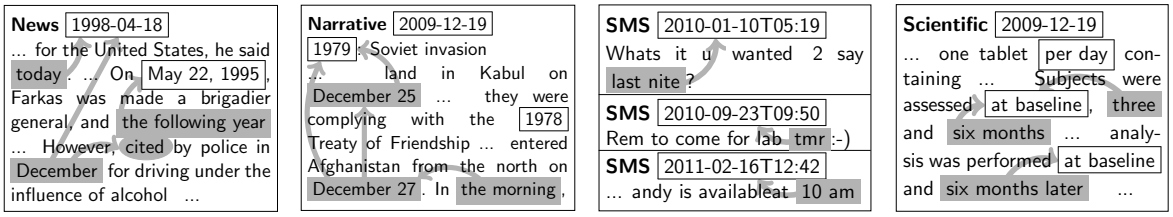
3.3.2 Characteristics of News-style Documents

Figure 3.2(a) shows excerpts of a document of the TimeBank corpus[17] as a representative of a typical news-style document. As will be further explained in the comparative corpus analysis in Section 3.3.7, the following features are characteristic for documents of the news domain.

A typical document of the news domain contains many date expressions. However, while explicit expressions (e.g., "May 22, 1995" in the given example) are rather rare, many of the occurring date and time expressions are either relative ("today", "the following year") or underspecified ("December"). Duration, time, and set expressions also occur but are much less frequent than date expressions.

In general, it is necessary to detect the correct reference time to correctly normalize relative and underspecified temporal expressions. In documents of the news domain, the document creation time can often be used as reference time, except for those relative expressions, which obviously refer to a previously mentioned expression (e.g., "the following year").

[16] `http://www.merriam-webster.com/dictionary/domain` [last accessed April 8, 2014].
[17] Excerpts are taken from document APW19980418.0210 of the TimeBank corpus.



(a) News-style. (b) Narrative-style. (c) Colloquial-style. (d) Autonomic-style.

Figure 3.2: Excerpts of documents of four domains showing domain-dependent characteristics. Explicit temporal expressions are marked with white boxes, underspecified and relative expressions with gray boxes. Gray arrows and ellipses indicate information required for normalization.

While it is sufficient to detect the reference time to normalize relative expressions, the relation to the reference time has to be determined to normalize underspecified expressions. For the normalization of "December" in the example, it is important to understand that "cited" refers to the past and that the citation is the event that took place in "December". Combining the information of the reference time and the relation "before" – due to the past tense – the expression can be normalized to "1997-12".

Table 3.2 summarizes the main characteristics of documents of all four domains considered in this work. In addition to providing an overview about the characteristics, this also allows a direct comparison between the characteristics of documents from the different domains.

3.3.3 Characteristics of Narrative-style Documents

An example of a typical document of the narrative domain is provided in Figure 3.2(b). It shows excerpts of the Wikipedia article "Soviet war in Afghanistan", which is part of the WikiWars corpus.

While narrative-style documents such as Wikipedia articles also contain many date expressions, there are important differences to news-style documents. On the one hand, explicit date expressions are much more frequent than in documents of the news domain (e.g., "1979" and "1978" in the example). Such expressions can be normalized without any context information once they are correctly extracted (cf. Section 2.3.2). However, relative and underspecified expressions occur also frequently ("December 25", "December 27", and "the morning"), and in contrast to in news-style documents, the document creation time is usually not eligible as reference time for such expressions. Thus, the reference time has to be detected in the text itself. For example, "1979" can be chosen as reference time for the expressions "December 25" and "December 27". Once the reference time is correctly identified, a promising assumption for documents of the narrative domain is that the text part between an explicit expression (the reference time) and an underspecified expression is typically structured chronologically. Note that the assumption is not that a whole article is structured chronologically but only (usually) short text parts between an underspecified expression and a previous expression determined as reference time.

Narrative-style documents often contain factual information and are typically quite long. Thus, they tend to have a rich temporal discourse structure, which makes the determination of the correct reference time for underspecified and relative temporal expressions more challenging (see also Mazur and Dale, 2010). Details about strategies how to address the different challenges of all the domains described here and in the following sections, will be provided in Section 3.3.8.

news domain	narrative domain	colloquial domain	autonomic domain
• many date expressions	• many date expressions	• many time expressions	• many date, duration, and set expressions
• many relative and underspecified expressions	• many explicit expressions	• hardly any explicit expressions	• many dates do not refer to real points in time
• reference time often document creation time	• reference time not document creation time	• reference time often document creation time	• reference time within local time frame
• detection of relation to reference time difficult	• long texts, rich temporal discourse structure	• missing context, no standard language	• reference times often no standard expressions

Table 3.2: Comparison of the main characteristics of documents of the domains considered in this work.

3.3.4 Characteristics of Colloquial-style Documents

Documents of our third domain of analysis are from the colloquial domain. Figure 3.2(c) shows excerpts of three short messages from the NUS SMS corpus[18] (Chen and Kan, 2013), which we used to create a colloquial-style temporally annotated corpus. In addition to short messages, colloquial-style texts can typically be found in tweets (see, e.g., Gimpel et al., 2011) or other textual social media content.

A main difference to documents of the news and narrative domains is that colloquial-style documents do not conform to standard language. There is a broad variety of spelling variations and word creations, e.g., the terms "night", "nite" (as in the first message in Figure 3.2(c)), "ni8", "nit" can all have the same meaning. The second message contains an additional example, "tmr" which means "tomorrow". Type errors and missing spaces (e.g., "availableat" in the third message) are also much more frequent than in non-colloquial documents. Thus, when processing documents of the colloquial domain, these issues have to be addressed. In particular, the broad variety of spellings have to be considered, e.g., by using synonym lists for relevant terms, i.e., terms with temporal meaning.

However, the style of language is not the only difference between colloquial-style documents and documents from other domains. Date expressions are less frequent while time expressions are used excessively, especially in short messages ("last nite", "10 am"). In addition, hardly any explicit date and time expressions occur but underspecified expressions, for which the reference time has to be detected. Fortunately, the time when a message was sent (i.e., the document creation time) is usually the reference time for underspecified and relative expressions in short messages and tweets. However, sometimes one might be faced with missing context information. "10 am" in the third example of Figure 3.2(c), for instance, could refer to the next "10 am" with respect to the sending time but if a previous message referred to another day, using the sending time as reference time would be the wrong guess for "10 am".

3.3.5 Characteristics of Autonomic-style Documents

The forth domain we are considering in our study contains so-called autonomic-style documents. In contrast to the other three domains, the name needs some further explanation: In our original study (Strötgen and Gertz, 2012b), we used scientific-style documents and thus called the domain *scientific domain*. However, while scientific documents are clearly part of the domain that we are describing, other documents may also be part of the domain. The main characteristic of documents of the autonomic domain is that

[18]The three excerpts are taken from the June 2011 version of the corpus and have the message ids 19314, 24333, and 27197.

they contain many temporal expressions, which cannot be normalized to some real point in time, but only according to some local or autonomic time frame. Thus, we consider all documents containing such a local time frame as being part of the *autonomic domain*. In the following, we present characteristics of such documents. These are also summarized in Table 3.2.

The scientific document in Figure 3.2(d) contains temporal expressions that refer to different points in time ("baseline", "three (month later)", "six month later"). These cannot be normalized to a real date according to a calendar but only with respect to the local or autonomic time frame. The expression "baseline" could be defined as "time point zero" of the document, while "three month (later)" and "six month (later)" refer to time points three and six month after this time point zero, respectively.

Examples of autonomic-style documents are scientific texts, e.g., documents describing clinical trials as the PubMed article[19] shown in Figure 3.2(d). Literary works[20] containing a local time frame also fall in the domain of autonomic-style documents. Note that there might be differences in good ways to identify so-called "time point zeros". While expressions such as "baseline" are frequent in scientific articles, determining possible "time point zeros" in literary text might be more challenging. In addition, longer articles may contain several local time frames, e.g., an autonomic time frame per chapter or paragraph.

While the key characteristic of autonomic-style documents is that they have a local time frame, we describe further characteristics that we analyzed in scientific documents (cf. Section 3.3.7). In contrast to documents of other domains, duration and set expressions are much more frequent. In addition, occurring date expressions are either explicit and refer to a real point in time or they are unresolvable according to usually used annotation and normalization standards. Thus, instead of normalizing expressions such as "six month later" as unknown point in time using "XXXX-XX-XX" as suggested by current annotation guidelines, we suggest to normalize them according to their local time frame to keep the temporal relations between expression such as "baseline", "three months (later)", and "six months (later)" shown in Figure 3.2(d). Detailed suggestions will be further explained in Section 3.3.8 when discussing strategies to address all kinds of domain-dependent challenges.

3.3.6 Corpus Creation

While there are several publicly available temporally annotated corpora containing news documents (e.g., the TimeBank corpus), and also a publicly available corpus containing narrative-style documents (the WikiWars corpus), neither a corpus containing colloquial nor a corpus containing autonomic documents have been available so far. In this section, we present the corpora, which we annotated in the context of this work: Time4SMS and Time4SCI.

Colloquial Corpus Creation

Although there are some SMS corpora publicly available, there are four main requirements for the SMS corpus to be applicable for publishing a temporally annotated SMS corpus: (i) it has to be freely available to allow others to reproduce the corpus, (ii) the language of the messages has to be English since for

[19] The excerpts are from the abstract of the paper "Supplementation with all three macular carotenoids: response, stability, and safety" by Conelli et al. (2011), `http://www.ncbi.nlm.nih.gov/pubmed/21979997/` [last accessed April 8, 2014].

[20] In the BMBF-funded interdisciplinary project heureCLÉA, we are currently working with narratologists from Hamburg University on the automatic identification of temporal phenomena in literary text. The basis of this work is to identify and normalize temporal expressions, i.e., to perform the task of temporal tagging on literary text documents.

developing a corpus for a new domain, English annotated corpora are most valuable for the research community, (iii) the corpus has to be large since the single messages are short and thus cannot contain many temporal expressions, and (iv) the document creation time (i.e., the time when the message was sent) has to be available for the messages. The availability of the sending time is crucial for normalizing underspecified and relative temporal expressions, which we expect to occur frequently in SMS texts.

Due to these requirements, we used the NUS SMS corpus (Chen and Kan, 2013) as basis of our colloquial corpus. However, the 2004 version of the corpus does not satisfy all our requirements, since these documents do not contain information about the sending time. Without the documents of the 2004 version, the corpus contains 28,268 messages (June 2011 version).[21] Due to privacy reasons, the developers of the corpus anonymized all short messages automatically and sensitive data were substituted by placeholders. Unfortunately, multi-digit numbers and some specific time information were part of this sensitive data. To overcome this problem, we replaced these placeholders of digits and times by some standard values in the original format.[22] Then, we randomly selected 1,000 documents as our SMS corpus called Time4SMS, in which we manually annotated all temporal expressions.

Scientific Corpus Creation

For the second new domain for our temporal analysis, we chose scientific documents. However, temporal expressions are not frequent in all kinds of scientific literature. A good representative of scientific documents with many temporal expressions are texts from the biomedical domain, e.g., publications about clinical trials. For selecting documents, we used PubMed,[23] which contains citations with abstracts and metadata such as publication dates of more than 20 million publications of the biological and biomedical domain. Using the PubMed search interface, we queried for "clinical trials" and downloaded abstracts and metadata of the 50 most recent publications. These documents form our scientific corpus called Time4SCI.

Annotation Procedure

As for the annotation of WikiWarsDE (cf. Section 3.4.5), we followed the developers of WikiWars (Mazur and Dale, 2010), i.e., we formatted the corpora in SGML, the format of the ACE TERN corpora. This makes it possible to evaluate temporal taggers on our newly annotated corpora using the publicly available TERN evaluation scripts.[24] The documents contain "DOC", "DOCID", "DOCTYPE", "DATETIME", and "TEXT" tags, and the document creation time (DATETIME) was set to the publication date being part of the metadata of each Pubmed article. The "TEXT" tag surrounds the text that is to be annotated.

For the annotation of temporal expressions, we used the TIDES TIMEX2 format (Ferro et al., 2005) with its attributes described in Section 3.2.1. Similar to Mazur and Dale (2010), we performed a three phase annotation process: (i) automatic pre-annotation, (ii) manual annotation with correcting wrong and adding missing expressions, and (iii) manual merging and validation of the annotations. For automatic pre-annotation, we used HeidelTime. Its output was then imported to the annotation tool Callisto[25] for the second annotation phase, the manual annotation and correction of wrong annotations.

[21] `http://wing.comp.nus.edu.sg/SMSCorpus/` [last accessed April 8, 2014].

[22] The NUS SMS corpus developers kindly provided their function to replace sensitive data, so that we were able to reproduce standard values for the placeholders in the original format.

[23] `http://www.ncbi.nlm.nih.gov/pubmed/` [last accessed April 8, 2014].

[24] We provide all necessary evaluation script at `http://code.google.com/p/heideltime/` [last accessed April 8, 2014]. In Section 3.6, further details about the evaluation will be provided.

[25] `http://mitre.github.io/callisto/` [last accessed April 8, 2014].

Note that the fact of using our own temporal tagger for automatic pre-annotation should be considered when comparing evaluation results of our temporal tagger with other taggers. However, as mentioned by Mazur and Dale (2010), this procedure is motivated by two facts. Firstly, "annotator blindness is reduced to a minimum" (Mazur and Dale, 2010), i.e., that annotators miss temporal expressions. Secondly, the annotation effort is reduced significantly since one does not have to create a TIMEX2 tag for the expressions already identified by the tagger (Mazur and Dale, 2010). In addition, this procedure is justifiable for our purpose because the main goals of creating the corpora are to study the differences between documents from different domains and to analyze domain-dependent challenges.

During the second annotation phase, the documents were examined for temporal expressions missed by the temporal tagger and annotations created by the temporal tagger were manually corrected. This task was performed by two annotators – although Annotator 2 only annotated the extents of temporal expressions. The more difficult task of normalization was performed by Annotator 1 only, since a lot of experience in temporal annotation is required. At the third annotation stage, the results of both annotators were merged and all normalizations of the expressions were checked and corrected by Annotator 1.

Finally, the annotated files, which contain in-line annotations, were transformed into the ACE APF XML format, a stand-off markup format used by the ACE evaluations. Thus, the Time4SMS and Time4SCI corpora can be made available in the same two formats as the WikiWars corpus and the evaluation tools of the ACE TERN evaluations can be used with the new corpora.

During the manual annotation process, we were faced with domain-specific difficulties. Due to many unresolvable temporal expressions in the scientific corpus, we suggest a new way to normalize these expressions. However, since the normalization of unresolvable expressions is one of the main challenges of temporal tagging autonomic documents, the details of this issue and how it can be addressed are described in the Section 3.3.7 and Section 3.3.8, respectively. Furthermore, in contrast to news- and narrative-style documents, it is very challenging to annotate colloquial and scientific text since deep domain knowledge is needed to fully understand such documents. For this, we regard our newly developed annotated corpora as preliminary versions of a gold standard.

3.3.7 Comparative Corpus Analysis and Domain-dependent Challenges

Using our two new corpora Time4SMS and Time4SCI as well as existing corpora of the news and narrative domains (TimeBank and WikiWars, respectively), we performed a comparative corpus analysis. During this analysis, we identified several challenges a domain-dependent temporal tagger is faced with. These challenges will be discussed in the following after presenting some statistics of the four corpora.

Corpora Statistics

Table 3.3 shows some statistics about the corpora. The documents of the Time4SMS corpus are very short. Although there may be longer colloquial texts, typical documents of this domain are short messages and tweets, which are both limited in their length. Thus, this characteristic will be observable for many colloquial documents. The Time4SCI documents are similar to the news documents in the TimeBank corpus with respect to the average length. Due to their shortness, documents in the Time4SMS corpus contain only a few temporal expressions. The average number of temporal expressions in the clinical-trial documents and in the news documents is comparable. In contrast, the narrative WikiWars documents are very long and contain many more temporal expressions than the documents in the other corpora.

corpus name	domain	doc	token	TIMEX	token/ doc	TIMEX/ doc
TimeBank	news	183	78,444	1,414	428.7	7.7
WikiWars	narrative	22	119,468	2,671	5430.4	121.4
SMS	colloquial	1,000	20,176	1,341	20.2	1.3
clinical-trial	scientific	50	19,194	317	383.9	6.3

Table 3.3: Statistics of temporally annotated corpora.

Although the Time4SMS and Time4SCI corpora are smaller than TimeBank and WikiWars with respect to the number of tokens, their sizes are sufficient to discover significant differences between the corpora resulting in several challenges for temporal tagging documents of different domains.

Types of Temporal Expressions

To identify challenges for temporal tagging documents of different domains, we analyze the temporal expressions occurring in the four corpora. The number of document creation times (DCTs) in the corpora equals the number of documents and thus, the percentage in corpora containing long documents with many temporal expressions is very low (WikiWars), but very high for those with short documents (Time4SMS). Since the DCT is usually easy to extract and to normalize or even directly provided as metadata about a document, we concentrate on temporal expressions occurring in the documents' texts in our further analysis.

In Figure 3.3, the frequencies of the four different types of temporal expressions (dates, times, durations, and sets) are depicted. In all four corpora, expressions of the type date are frequent. However, there are significant differences between the four domains with temporal expressions being of the type "date" covering between 40% of the temporal expressions in the Time4SCI corpus and almost 90% in the narrative corpus. In contrast, time and set expressions are only frequent in the colloquial and clinical trial corpora, respectively. Duration expressions are well-covered in all corpora although at a lower level than date expressions. Furthermore, duration expressions are much more frequent in the scientific corpus than in the other three corpora.

Due to the differences in the distribution of types of temporal expressions in the corpora of the different domains, the following problem becomes obvious: when developing a temporal tagger on one domain only, e.g., on the news domain as are most existing systems, this may result in a worse coverage on the other domains since not all types of expressions may be covered very well. For example, it would be possible to extract more than 80% of the temporal expressions from the news and narrative corpora with a temporal tagger that only extracts date expressions. However, on the colloquial and scientific corpora only about 50% and 40% of the expressions would be extractable at all. Thus, the first challenge for temporal tagging on multiple domains can be formulated as follows:

> **Challenge 1: Broad Coverage.**
> There is a need that all four types of temporal expressions (dates, times, durations, and sets) are well covered by a temporal tagger.

Analyzing the distribution of date, time, duration, and set expressions already shows first significant differences between documents from different domains. However, these temporal expressions, and in particular date and time expressions, can occur in different ways (cf. Section 2.3.2). In the following, we thus analyze date and time expressions in more detail.

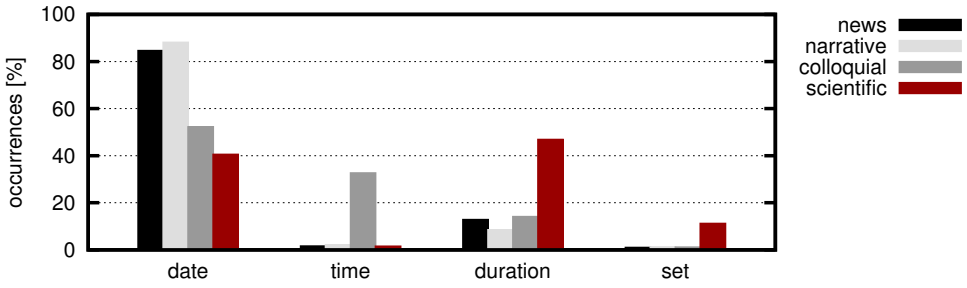


Figure 3.3: Distribution of types of temporal expressions in the four corpora.

Characteristics of Date and Time Expressions

Temporal expressions of the types date and time can either be explicit, implicit, relative, or underspecified. In addition, in some types of documents, so-called unresolvable temporal expressions occur as will be detailed below. The occurrence type directly results in different challenges for temporal tagging, especially for the normalization of temporal expressions.

Figure 3.4 shows the occurrence types of date and time expressions in the four corpora. The easy to normalize explicit temporal expressions are frequent in the WikiWars corpus (> 50%) while they rarely occur in the colloquial corpus (< 0.5%). Implicit expressions are rare in all the four corpora. However, to extract and normalize the occurring implicit expressions, the temporal tagger requires additional knowledge resources. For example, to extract and normalize holidays and expressions such as "D-Day", they have to be known by the tagger in the same way as usual temporal words such as names of months. Thus, the second challenge for a temporal tagger can be described as follows:

> **Challenge 2: Resources for Implicit Expressions.**
> If the documents of a specific domain contain many implicit expressions, there is the need to easily add resources to extract and normalize them.

To normalize relative and underspecified temporal expressions, e.g., "next Monday" or "November" in phrases such as "In November", the temporal tagger has to identify the reference time of the corresponding expressions. In news-style and colloquial-style documents, the identification of the reference time is relatively simple since it is often the document creation time (DCT) or sending time, respectively. In narrative-style documents, almost always the reference time has to be determined in the documents' texts. Thus, the third challenge of a temporal tagger can be formulated as follows:

> **Challenge 3: Reference Time Identification.**
> To be able to normalize relative and underspecified temporal expressions, the temporal tagger has to identify the correct reference time.

In Figure 3.4, we distinguish between those relative and underspecified expressions for which the document creation time is the reference time (ref=dct), and those for which the document creation time cannot be used as reference time (ref≠dct). In the news and colloquial corpora, there are about 78% and 86% of the date and time expressions either relative or underspecified with the document creation time being the reference time. In contrast, only for 10% of the expressions in the news corpus, the reference time has to be identified in the text. In the colloquial corpus, such expressions did not occur at all.

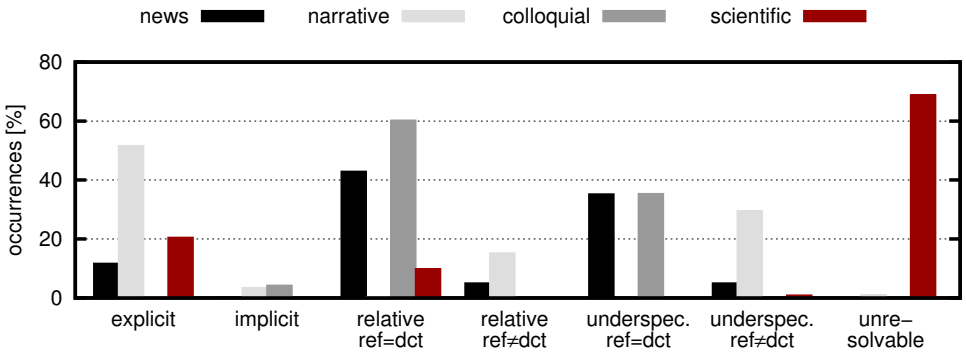


Figure 3.4: Characteristics of time and date expressions in the analyzed corpora (ref: reference time; dct: document creation time).

In the narrative-style corpus, almost 45% of the date and time expressions are relative or underspecified and their reference time has to be detected in the documents' text, while less than 0.5% of the expressions have the document creation time as reference time. Furthermore, due to the large number of temporal expressions in the documents of the narrative corpus (cf. Table 3.3), the temporal discourse structure is more complex, i.e., the reference time identification tasks is even more challenging in narrative-style documents. In the scientific corpus, relative and underspecified expressions are rare, but if they occur their reference time is usually the DCT.

In summary, it is more challenging to identify the reference time in narrative-style documents than in the other domains since it has to be determined in the text and is usually not the DCT. Thus, to address Challenge 3, a temporal tagger should apply domain-dependent strategies to identify the reference time of relative and underspecified expressions.

In contrast to normalizing relative expressions, for the normalization of underspecified expressions, it is not sufficient to identify the reference time, but the relation to the reference time is also needed. This is a challenging task independent of the domain.

> **Challenge 4: Identification of the Relation to the Reference Time.**
> To normalize underspecified temporal expressions correctly, the relation to the reference time has to be identified in addition to the reference time.

As will be detailed in the next section, if the DCT is the reference time of an underspecified expression, tense information about the sentence in which the expression occurs may be helpful. If the tense information cannot be identified, e.g., several SMS texts in the colloquial corpus do not contain any verb at all, the normalization will be even more challenging and the relationship between the underspecified expression and its reference time has to be guessed. While news documents often describe events that already happened, the analysis of the Time4SMS corpus suggests that SMS messages tend to refer to upcoming events. If the reference time is not the DCT, one may assume that there is a partially chronological order in the text, i.e., that an underspecified expression refers to a point in time after a previously mentioned reference time.

Thus, to address Challenge 4, domain-dependent strategies are needed in particular for processing news-, narrative-, and colloquial-style documents in which underspecified temporal expressions are quite frequent. These strategies will be presented in Section 3.3.8.

Non-standard Language

In colloquial texts, further challenges arise, which hardly occur in neither news, narrative, nor scientific documents. These challenges include (i) a broad variety of spelling variations and word creations, (ii) type errors, and (iii) missing spaces. These issues will be explained in more detail below and can be summarized as follows:

> **Challenge 5: Coping with Non-standard Language.**
> In some domains, non-standard language issues may occur frequently and have to be considered by the temporal tagger.

Examples of the first type of non-standard language, i.e., spelling variations and word creations, are synonyms for the word "night", which we identified in the colloquial corpus Time4SMS: "night", "nite", "nit", and "ni8". There are word creations and spelling variations for all kinds of expressions. To be able to perform temporal tagging on colloquial texts, at least the synonyms for temporally relevant terms have to be known by a temporal tagger. Otherwise, all temporal expressions containing non-standard language words could not be extracted correctly.

The other two issues mentioned above – type errors (e.g., "mornimg") and missing spaces (e.g., "todaygot") – also occur frequently in the colloquial corpus. However, performing spelling correction on colloquial text documents is a non-trivial task due to the intentional usage of non-standard language words. In the next section, we suggest some strategies to deal with these problems and to address these challenges. In general, these issues usually occur only in colloquial documents and should thus be handled by a temporal tagger if colloquial text is processed.

An additional challenge occurring in the documents of the Time4SMS corpus is that required context information may have been mentioned in previous messages but cannot be accessed for the normalization. For instance, in the third example shown in Figure 3.2(c) (page 49), we can only assume that the "10 am" mentioned in the message ("andy is availableat 10 am") refers to "10 am one day after the document creation time". While it seems to be likely, depending on the previous parts of the conversation, the expression could also refer to "10 am" at another day.

This issue can occur in every corpus containing parts of conversations. However, this challenge can only be addressed if the conversation (e.g., several SMS that build a conversation) is processed as a single document. Thus, this challenge is not a challenge that can be addressed by the temporal tagger itself, but may be addressed during corpus preprocessing.

Unresolvable Temporal Expressions due to Local Time Frames

While Challenge 5 is mostly relevant for the colloquial corpus, we identified another challenge that is mainly relevant for the scientific corpus since it affects many temporal expressions in this corpus. Often, these documents contain their own time frame, e.g., the beginning of a clinical trial. This results in the fact that although there are several temporal expressions in these documents, many of them cannot be normalized to some real point in time. For example, the document shown in Figure 3.2(d) (page 49) contains the expression "six months later". However, it is not intended that this expression can be grounded to a real point in time, i.e., to a specific date. The important information is that the expression refers to the point in time six month after the baseline.

	challenge	description
1	broad coverage	All types of temporal expressions should be well covered by the tagger.
2	implicit expressions	Resources for implicit temporal expressions should be easily extensible.
3	reference time	For relative and underspecified temporal expressions, domain-dependent strategies to detect the reference time are required.
4	relation to reference time	For underspecified temporal expressions, domain-dependent strategies to determine the relation to the reference time are required.
5	non-standard language	Spelling variations, word creations, and type errors may have to be addressed when processing specific documents, e.g., colloquial texts.
6	local time frames	Some documents contain local time frames resulting in problems to normalize relative expressions with respect to current annotation guidelines.

Table 3.4: Six challenges that have to be addressed by a cross-domain temporal tagger.

If temporal expressions cannot be normalized to some real point in time, the annotation guidelines of TimeML suggest that they are normalized in an underspecified way. For instance, if the expression refers to a date of the granularity day, the normalized value will be XXXX-XX-XX. Then, however, the temporal relation between "baseline" and "six months later" is lost. Thus, instead of normalizing such expressions to unspecific values, we suggest to create a local time frame for each document, and to normalize relative expressions with respect to the local time frame. In the next section, we will describe how such a local time frame can be created and how relative expressions can then be normalized.

As shown in Figure 3.4, almost 70% of the date and time expressions in the scientific corpus are unresolvable expressions. Although such expressions hardly occur in the other analyzed corpora, dealing with this type of expressions is important when processing documents containing local time frames.

> **Challenge 6: Local Normalization of Unresolvable Temporal Expressions.**
> In some domains, unresolvable time and date expressions occur. These cannot be normalized to a global point in time and should be normalized with respect to a local time frame.

Note that Challenge 6 is not only typical in scientific documents but also in literary documents and (fictional) narrative stories, which may contain several temporal expressions related to each other in a local time frame only. Thus, as already pointed out above, we call the domain containing documents with local time frames *autonomic domain* rather than scientific domain.

Summarizing the Challenges for a Domain-sensitive Temporal Tagger

In summary, there are several challenges for temporal tagging. While some of them are domain-independent, others arise only when processing specific domains. For instance, identifying the reference time of relative and underspecified temporal expressions is necessary to normalize such expressions independent of the domain. However, due to the different characteristics of the documents from different domains, it is necessary to tackle the challenges in a domain-dependent manner.

3.3.8 Strategies to Address Domain-dependent Challenges

In Table 3.4, the six challenges described in the previous section are summarized. In the following, we show how they can be addressed by a temporal tagger applying domain-dependent strategies.

Guarantying Broad Coverage of a Temporal Tagger

Challenge 1, i.e., that a temporal tagger should cover temporal expressions of all types adequately, can be tackled if a temporal tagger is developed based on data of all domains that shall be processed. Either a machine-learning based temporal tagger should be trained using training data of all domains or the rules of a rule-based temporal tagger should be developed based on examples of all domains.

Providing Extensibility of Resources for Implicit Expressions

The second challenge listed in Table 3.4, i.e., that implicit temporal expressions can be integrated easily, can be solved if the architecture of a temporal tagger supports the simple integration of additional resources. While the vocabulary of standard temporal expressions is limited, e.g., based on numbers and names of months and days, the vocabulary of implicit expressions is potentially unlimited. Thus, to extract and normalize these expressions, the temporal tagger should have access to resources in a modular way.

Detecting Reference Times of Relative and Underspecified Expressions

While the first two challenges do not require domain-dependent strategies, the third challenge listed in Table 3.4 – the identification of the reference time of relative and underspecified expressions – should be addressed differently depending on the domain of the documents that are processed.

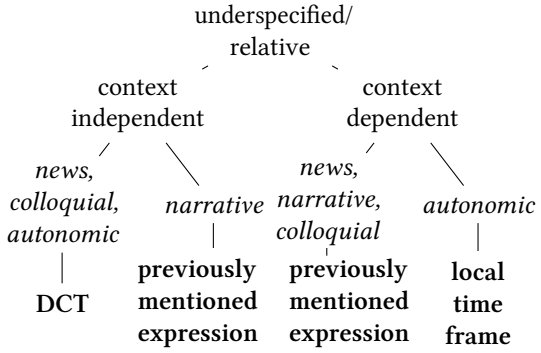
On the one hand, as already described, the reference time of underspecified and relative temporal expressions in news documents is often the document creation time (DCT). The same is true for such expressions in colloquial and scientific documents. On the other hand, narrative-style documents usually do not contain any relative and underspecified temporal expressions, for which the reference time is the DCT. In contrast, another temporal expression has to be identified as reference time. Although identifying the correct reference time in narrative-style documents is sometimes difficult and tricky, a simple strategy is to use the previously mentioned temporal expression of the required granularity as reference time.

A promising general strategy to detect the correct reference time of relative and underspecified expressions is depicted in Figure 3.5(a). Note that there are some relative temporal expressions such as "two days later" for which the reference time has to be identified in the text independent of the domain of the document (context-dependent expressions). As shown in Figure 3.5(a), the previously mentioned expression can be used as reference time for such context-dependent expressions on all domains except the autonomic domain where the local time frame has to be considered.

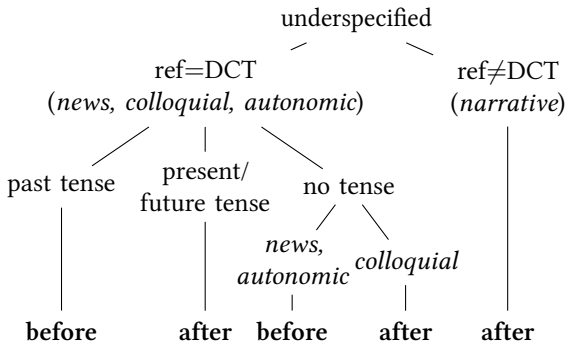
Note that while the strategy for reference time detection depicted in Figure 3.5(a) is a promising approach and often works well, some temporal expressions in text documents are not reliable candidates for reference times. For instance, temporal expressions describing background information often tend to be not eligible as "1978" in the example depicted in Figure 3.2(b) (page 49). To determine if a temporal expression refers to background information is rather difficult. However, it may already be useful to remove attributive temporal expressions from the set of candidates. Then, the suggested strategy depicted in Figure 3.5(a) could be used with ignoring unlikely candidates, e.g., the "1978" in Figure 3.2(b).

Detecting Relations to Reference Times of Underspecified Expressions

For underspecified temporal expressions, in addition to the reference time, the relation to the reference time has to be detected (Challenge 4 in Table 3.4). If the reference time is the document creation



(a) Domain-dependent strategies (italics) to identify the reference time (bold).



(b) Domain-dependent strategies (italics) to identify the temporal relation to the reference time (bold).

Figure 3.5: Strategies to identify the reference time (a) and the temporal relation to the reference time (b).

time (DCT), a promising approach is to identify the tense of the sentence. While past tense indicates that the relation between the underspecified expression and the DCT is "before", future tense (and usually also present tense) indicate that the relation is "after". However, in some cases, there is no tense in the sentence and thus the relation has to be guessed. Then, we suggest a domain-dependent strategy. As described in Section 3.3.7, news are more likely to refer to past events, while colloquial documents such as SMS and tweets tend to refer to future events.

If the reference time is not the DCT, which is the case if an underspecified expression occurs in narrative-style documents, a promising assumption is that the expressions occur chronologically in the document. Note that this assumption is not made in general to all temporal expressions in a document but only concerns the underspecified expression that is under consideration and the previously mentioned expression, which is determined as its reference time. The general strategy to determine the temporal relation between an underspecified expression and its reference time is depicted in Figure 3.5(b).

In summary, the normalization of relative and underspecified temporal expressions can be addressed by applying domain-dependent strategies for challenges 3 and 4.

Coping with Non-standard Language

The fifth challenge listed in Table 3.4 is that a temporal tagger should deal with non-standard language. In contrast to the previous challenges, this issue mainly occurs in colloquial text documents and thus only has to be tackled when processing documents from this domain.

For spelling variations and word creations that refer to temporal expressions, e.g., "tmr" for "tomorrow", we suggest to add the synonyms to the pattern resources of the temporal tagger. Then, non-standard language words can be extracted and normalized by the temporal tagger as regular words. Details how we realized this strategy for our temporal tagger HeidelTime will be presented in Section 3.5.7.

More difficult but also frequently occurring challenges are typing errors. As already mentioned above, standard methods to address this issue are not suitable since colloquial text documents usually contain intended non-standard language words. Thus, a regular spelling correction tool cannot be used to correct

type errors since intended non-standard language words would be "corrected" as well. Therefore, we suggest to tackle this issue by searching for inexact patterns. Depending on the length of an expression that is to be matched, one could specify a threshold and calculate edit distances, e.g., the Levenshtein distance (see, e.g., Manning et al., 2008: p.58). If the edit distance is below the threshold, inexact matches could be extracted and normalized according to the edited expression.

A third variation of Challenge 5 are missing spaces between a temporal expression and the previous or next token. While this issue could be seen as a variant of type errors, we suggest to tackle missing spaces by removing the generally used constraint that temporal expression have to begin and end with the beginning and ending of a token, respectively. To avoid many false matches, one may want to validate that the whole token is not an existing word, e.g., by using a dictionary. This would allow to extract "today" in the expression "todaygot" but avoid to extract "May" in the expression "Maybe", for instance.

Since these strategies for coping with non-standard language issues include modifications to the language resources of a temporal tagger, we suggest to handle colloquial language as a separate language. For instance, when a temporal tagger is supposed to process English colloquial text documents, the tagger could contain language resources for English and English-colloquial. This also guarantees that the tagging quality on standard language documents will not suffer due to non-standard language synonyms.

Normalizing Unresolvable Temporal Expressions with respect to the Local Time Frame

The last challenge listed in Table 3.4 is a complex challenge mainly occurring in documents of the autonomic domain, i.e., in the documents of the scientific corpus in our comparative corpus analysis. As explained above, it is important that unresolvable temporal expressions are not normalized to unspecific points in time (e.g., XXXX-XX-XX), but with respect to a local time frame, if possible.

In order not to lose information about the relations between temporal expressions, we suggest to normalize such unresolvable expressions according to a local time frame, i.e., a time point zero that has to be detected in the document. Note that this is only possible if the annotation standard for temporal expressions (e.g., TimeML's TIMEX3) is extended.

We suggest to start using the local semantics of temporal expressions as defined by Mazur and Dale (2011), e.g., "one day later" is normalized to "+0000-00-01". However, we suggest to combine local semantics of expressions with local time frames of documents. Then, in cases of chains of relative expressions, the semantics can be accumulatively added. For instance, a document about a clinical trial may contain the following text *"... baseline ... two days later ... one day later"*. Then "two days later" could be normalized to "TPZ+0000-00-02" and "one day later" to "TPZ+0000-00-03" referring to two and three days after the time point zero (TPZ) referred to by the expression "baseline".

While we introduced the concept of a *time point zero* per document in (Strötgen and Gertz, 2012b), we extend this concept here in such a way that a document may contain several time point zeros. Assuming a longer document, e.g., a document containing multiple clinical trials, it is likely that there are several time point zeros, i.e., time points which are required as anchors for the normalization of other temporal expressions. These anchor time points have to be identified and normalized non-ambiguously, so that relative temporal expressions can be normalized to the correct time point zero. Thus, we suggest to number consecutively the time point zeros of a document, e.g., using "TPZ0", "TPZ1", etc. Note that this extension does not require any changes in the annotation of the documents of the clinical trial corpus (Time4SCI) since these contain at the most one time point zero per document due to their shortness.

3.3.9 Summary

In summary, it is crucial for a temporal tagger that is intended to process documents of different domains to be developed in such a way that a domain-sensitive temporal tagging can be performed. While most previous approaches to temporal tagging dealt with documents from the news domain only, we developed our cross-domain temporal tagger HeidelTime, which distinguishes between four domains (news-, narrative-, colloquial-, and autonomic-style documents) and takes care of the challenges for cross-domain temporal tagging by deploying most of the strategies described above.

In Section 3.5, we will present the details of HeidelTime and in Section 3.6 we perform a detailed evaluation including a cross-domain evaluation demonstrating the usefulness of HeidelTime's cross-domain temporal tagging strategies. Before doing so, we will address the next open issue described in Section 3.2.6, i.e., multilingual temporal tagging.

3.4 Multilingual Temporal Tagging

Most of the research on temporal tagging done so far is for processing English text documents. There are barely any multilingual temporal taggers supporting more than two languages. While the state-of-the-art in English temporal tagging was already described in the previous section, we present the state-of-the-art in temporal tagging of other languages in the following. First, in Section 3.4.1, we give an overview of the languages that are subject of analysis in this work, i.e., languages, which can already be processed with our temporal tagger HeidelTime. Then, research challenges, temporally annotated corpora, and existing temporal taggers for languages other than English will be surveyed (Section 3.4.2 to Section 3.4.4).

Since there have not been annotated corpora for all the languages addressed in this work, we manually annotated some non-English corpora to be able to present evaluation results of HeidelTime later in this work. These corpora and their development are described in Section 3.4.5. Finally, in Section 3.4.6, we present the most important language-specific characteristics and challenges we detected during our work.

3.4.1 Languages Addressed in this Work

As will be detailed in Section 3.5, HeidelTime's current version (version 1.5) supports eight languages: English, German, Dutch, Spanish, Italian, French, Arabic, and Vietnamese. While HeidelTime's Dutch and French capabilities have been developed by other researcher (van de Camp and Christiansen, 2012; Moriceau and Tannier, 2014), we addressed English, German, Spanish, Italian, Arabic, and Vietnamese at our institute in the context of this work. The reasons for this set of languages are as follows:

- Several of the concepts introduced in the following chapters of this work rely on temporal information and are language-independent. To demonstrate their language-independence, it was also important to address the task of temporal tagging for more than one language.
- *English* is without any doubt the most important language to address when performing NLP research. In addition, annotated corpora are available and there is a lot of research dealing with English which can be used for comparison.
- *German* is the native language of the author of this work and there is an active NLP research community dealing with processing German texts. However, no temporal tagger for processing German texts has been available so far.

- For some further languages, temporally annotated corpora are publicly available and research challenges have been organized. In contrast to other languages for which annotated corpora are also available, the author of this work has basic language skills in *Spanish* and *Italian*. These are helpful when developing a rule-based system.
- Finally, *Arabic* and *Vietnamese* HeidelTime resources have been developed with the help of native speaking researchers being part of our research group (Strötgen et al., 2014a). For both languages, HeidelTime is the first publicly available system for performing the full task of temporal tagging.

Thus, despite the fact that some of the addressed languages are not the typically addressed languages in NLP research, we made important contributions to the research community. We hope that research and tools relying on temporal information will now be applied or extended in a multilingual way. Furthermore, as will be detailed in Section 3.7, further languages will be supported by HeidelTime in the future.

In the following sections, when presenting the state-of-the-art in non-English and multilingual temporal tagging, we will focus on the languages addressed in the context of this work but will also point to research on further languages.

3.4.2 Research Competitions for other Languages than English

As described in Section 3.2.2, there have been several research competitions dealing with temporal tagging. While the first competitions in the context of the Message Understanding Conferences, MUC-6 (Grishman and Sundheim, 1995) and MUC-7 (Chinchor, 1997), only addressed English temporal information extraction, later challenges were organized in a multilingual way or addressed languages other than English. In the following we give an overview of these challenges.

ACE TERN Competitions

In the first ACE TERN contest in 2004, the task of temporal tagging of Chinese was offered in addition to English temporal tagging, and both languages have been addressed by participants (see, e.g., Negri and Marseglia, 2004; Mingli et al., 2005). However, in the competition, the normalization of temporal expressions was only considered for English.

In 2005, ACE tasks for three languages have been organized, namely for English, Chinese, and Arabic. However, although the ACE Multilingual 2005 training corpus (cf. Section 3.4.3) contains temporally annotated data for all three languages, the TERN task was not carried out for Arabic according to the ACE 2005 evaluation plan.[26]

Finally, for the third ACE contest in 2007 a forth language was added (Spanish). According to the ACE 2007 evaluation plan, the TERN task was planned to be carried out in all four languages.[27] According to the official evaluation results, there have been participants for temporal tagging of English, Chinese, and Spanish.[28] While four teams addressed the task of English temporal tagging, there was only one participating team for Spanish and one for Chinese temporal tagging.

[26] See `http://www.itl.nist.gov/iad/894.01/tests/ace/2005/doc/ace05-evalplan.v3.pdf` [last accessed April 8, 2014] and also Mazur (2012).

[27] `http://www.itl.nist.gov/iad/mig/tests/ace/ace07/doc/ace07-evalplan.v1.3a.pdf` [last accessed April 8, 2014].

[28] `http://www.itl.nist.gov/iad/mig/tests/ace/2007/doc/ace07_eval_official_results_20070402.html` [last accessed April 8, 2014].

TempEval Competitions

While the first TempEval challenge in 2007 did not consider temporal tagging (Verhagen et al., 2007), this task was part of the second TempEval challenge organized in 2010 (Verhagen et al., 2010). For the first time, the full task of temporal tagging, i.e., the extraction and the normalization of temporal expressions, was offered in several languages: English, Spanish, Italian, French, Chinese, and Korean. However, most of the participants addressed the task of temporal tagging only for English, so that there were eight teams participating in the task of temporal tagging of English. While two teams also performed the task of Spanish temporal tagging, there were no participants addressing the other four languages.

In the third TempEval challenge (UzZaman et al., 2013), the number of languages was reduced and only tasks in English and Spanish have been organized. Again, there have been many more participating teams submitting approaches for English (9 teams) than for Spanish (2 teams) temporal tagging.

The EVALITA Competition

Finally, there has been a research challenge on temporal tagging of Italian text documents organized in 2007. The EVALITA TERN competition (Bartalesi Lenzi and Sprugnoli, 2007) offered the full task of temporal tagging and there have been four participants.

Summary

While there have been efforts to organize temporal tagging research challenges for languages other than English, these have not received as much attention as English challenges. On the one hand, sometimes only subtasks were organized for non-English languages. On the other hand, if the full temporal tagging tasks have been organized, not many researchers addressed non-English temporal tagging.

3.4.3 Non-English Corpora

There are several non-English corpora containing annotations of temporal expressions. They have been developed either in the context of one of the challenges described above or as part of a temporal annotation project. In the following, we will survey these corpora with a focus on corpora with documents of the languages addressed in this work. Table 3.5 shows an overview of non-English corpora – including non-English corpora we developed in the context of this work. Their development will be detailed in Section 3.4.5.

ACE Multilingual 2005 Training Corpus (Arabic and Chinese)

In addition to English documents, the ACE Multilingual 2005 training corpus[29] also contains temporally annotated Arabic and Chinese documents. However, for both languages, only extent information and no normalization information is provided.

The Arabic and Chinese parts of the corpus are made up of newswire (40%), broadcast news (40%), and weblog (20%) texts, and the TIDES TIMEX2 standard is used for annotation. Although the Arabic part of the corpus consists of 433 documents, only 403 documents are adjudicated after a dual annotation phase

[29] The ACE Multilingual 2005 training corpus is released by the Linguistic Data Consortium (LDC), catalog number LDC2006T06; `http://www.ldc.upenn.edu/` [last accessed April 8, 2014].

and thus considered as "high-quality gold standard" by the developers (Walker et al., 2006). These 403 documents contain 2,302 annotated temporal expressions. In the Chinese part of the corpus, there are 687 documents. 633 of them are adjudicated after a dual annotation phase and contain 4,986 TIMEX2-annotated temporal expressions.

The TempEval-2 Data Sets (Spanish, French, Italian, Chinese, and Korean)

The TempEval-2 corpus consists of English, Spanish, French, Italian, Chinese, and Korean training and test sets.[30] All parts of the corpus are annotated following the TimeML annotation guidelines, i.e., temporal expressions are annotated with TIMEX3 tags. While the development of this multilingual corpus was a big step towards multilingual research on temporal tagging (and temporal annotation in general), the organizers stated that one should not place too high expectations on the quality of the annotations in the non-English parts of the corpus:

> "It should be noted that for some languages the annotations are a bit experimental. For all languages but English, and to a lesser extent Italian, the TempEval-2 annotation was the first temporal annotation of this kind." (TempEval-2 release notes, Verhagen, 2011)

Fortunately, the annotation efforts have been pursued and resulted in several high quality language resources for some of the languages as described in the following. For the evaluation of HeidelTime, we only used the Italian part of the TempEval-2 corpus since for the other languages of interest, improved versions of the annotations are available. The Italian training and the test sets contain 51 and 13 documents with 523 and 126 annotated temporal expressions, respectively.

The TempEval-3 Data Sets (Spanish) and the Spanish TimeBank

While the annotations of the TempEval-2 data have been used as basis for the Spanish TimeBank, in the context of the TempEval-3 competition, the final Spanish TimeBank corpus (Saurí and Badia, 2012) has been developed. For the TempEval-3 competition (UzZaman et al., 2013), the corpus was split into a training set and a test set containing 175 and 35 documents with 1,094 and 198 annotated temporal expressions in the text parts of the documents, respectively. The corpus is publicly available.[31]

The EVALITA I-CAB Corpus

The I-CAB corpus[32] is annotated following the TIDES TIMEX2 annotation guidelines with minor modifications to address Italian language specificities (Magnini et al., 2006). The corpus was split into a training and a test set for the EVALITA competition (Bartalesi Lenzi and Sprugnoli, 2007). They contain 335 and 190 documents with 2,901 and 1,652 annotated temporal expressions, respectively. Note that the corpus contains news articles but – in contrast to most of the other temporally annotated corpora – the documents are from different newspaper genres. This results in a broad variety of temporal expressions as we will detail in the error analysis of HeidelTime's evaluation results (cf. Section 3.6.8).

[30] The corpus and evaluation scripts are publicly available, `http://www.timeml.org/site/timebank/tempeval/tempeval2-data.zip` [last accessed April 8, 2014].

[31] The Spanish TimeBank corpus is distributed by the Linguistic Data Consortium, catalog number LDC2012T12, `http://www.ldc.upenn.edu/` [last accessed April 8, 2014]; the Spanish TempEval-3 data sets can be downloaded from the TempEval-3 website: `http://www.cs.york.ac.uk/semeval-2013/task1/` [last accessed April 8, 2014].

[32] Available upon request, `http://ontotext.fbk.eu/i-cab/download-icab.html` [last accessed April 8, 2014].

language	corpus name	subset	domain	annotation standard[1]	documents	temporal expressions
Spanish	Spanish TimeBank		news	TIMEX3	210	1,322[2]
Spanish	Modes-TimeBank			TIMEX3	102	892[3]
Spanish	TempEval-2	training	news	TIMEX3	173	1,092
Spanish	TempEval-2	test (all)	news	TIMEX3	35	199
Spanish	TempEval-3	training	news	TIMEX3	175	1,094
Spanish	TempEval-3	test	news	TIMEX3	35	198
Italian	Ita-TimeBank[4]		news	TIMEX3		
Italian	TempEval-2	training	news	TIMEX3	51	523
Italian	TempEval-2	test (all)	news	TIMEX3	13	126
Italian	I-CAB	training	news	TIMEX2	335	2,901
Italian	I-CAB	test	news	TIMEX2	190	1,652
French	French TimeBank		news	TIMEX3	108	533
French	TempEval-2	training	news	TIMEX3	83	206
French	TempEval-2	test (all)	news	TIMEX3	15	83
Portuguese	TimeBankPT	training	news	TIMEX3	162	1,244[5]
Portuguese	TimeBankPT	test	news	TIMEX3	20	165[5]
Romanian	Rom. TimeBank		news	TIMEX3	183	1,414[6]
Korean	TempEval-2	training	news	TIMEX3	18	287
Korean	TempEval-2	test (all)	news	TIMEX3	4	95
Chinese	TempEval-2	training	news	TIMEX3	44	746
Chinese	TempEval-2	test (all)	news	TIMEX3	15	190
Chinese	ACE 2005 training		news	TIMEX2*	633	4,986
Arabic	ACE 2005 training		news	TIMEX2*	403	2,302
Arabic	ACE 2005 training	training-203	news	TIMEX2*	203	1,137
Arabic	ACE 2005 training	test-150	news	TIMEX2*	150	904
Arabic	ACE 2005 training	test-50	news	TIMEX2*	50	261
Arabic	ACE 2005 training	test-50-star	news	TIMEX3	50	298
German	WikiWarsDE		narratives	TIMEX2	22	2,240
Vietnamese	WikiWarsVN		narratives	TIMEX3	15	226

Table 3.5: Overview of non-English corpora annotated with temporal expressions grouped by language.
[1]Corpora marked with * do not contain annotated normalization information.
[2]According to Saurí and Badia (2012).
[3]According to Guerrero Nieto et al. (2011).
[4]The Ita-TimeBank is not yet released. Caselli et al. (2011) describe statistics on a subset.
[5]According to Costa and Branco (2012).
[6]According to Forascu and Tufis (2012).

The Italian TimeBank

While the developers of the Italian Timebank corpus (Ita-TimeBank) described the annotation guidelines and specifications developed in the context of the corpus creation (Caselli et al., 2011), the full corpus itself has not been released yet.[33]

The French TimeBank

Although the French HeidelTime resources were developed by other researchers (Moriceau and Tannier, 2014), we will also present HeidelTime's evaluation results for processing French. For this, we will use the French TimeBank corpus (Bittar et al., 2011). The temporal expressions in the corpus are annotated using TIMEX3 tags, and the 108 documents contain 533 annotated temporal expressions. The documents in the corpus all come from the news domain, although from different sub-genres such as national and international news (Bittar et al., 2011). The corpus is publicly available.[34]

Further Publicly Available Corpora

Mainly recently, there is a new interest in temporal annotation for languages other than English. In addition to the corpora surveyed above, there are some more corpora containing documents of other languages than those addressed in the context of this work. For instance, the Romanian TimeBank (Forascu and Tufis, 2012) and the Portuguese Timebank (Costa and Branco, 2012) have been made available, recently. In addition, the ModeS TimeBank corpus (Guerrero Nieto et al., 2011; Guerrero Nieto and Saurí, 2012) contains documents from the 17th and 18th centuries written in Modern Spanish. All three corpora contain TIMEX3 annotations for temporal expressions and are publicly available.[35]

3.4.4 Non-English Temporal Taggers

There are three methods to develop non-English and multilingual temporal taggers. Either a system is developed for another language solely relying on resources of the target language (e.g., annotated corpora), or an existing approach for one language is extended to process further languages. In the latter case, one can further distinguish if porting from one language to another one is performed manually or by (semi-)automatic methods. While some of the taggers described in the following have already been presented in Section 3.2.5 where English temporal taggers were surveyed, others are introduced here.

Chronos – Manual Adaptation to Target Language

Chronos is a rule-based, TIMEX2-compliant temporal tagger originally developed for English temporal tagging (Negri and Marseglia, 2004), and extended to process Italian (Negri, 2007). This extension was performed by manually creating language resources such as rules for the target language. Note, however, that this manual extension from one language to another one can be quite fast if the developer is familiar with the target language and the system's architecture (Negri, 2007). In Section 3.2.5, Chronos' system architecture was already detailed.

[33] `http://www.celct.it/projects/it-timeml.php` [last accessed April 8, 2014].
[34] `http://www.linguist.univ-paris-diderot.fr/~abittar/french-timebank/` [last accessed April 8, 2014].
[35] The Romanian TimeBank corpus: `http://www.meta-share.eu/` [last accessed April 8, 2014]; the Portuguese TimeBank: `http://nlx-server.di.fc.ul.pt/~fcosta/TimeBankPT/` [last accessed April 8, 2014]; the ModeS TimeBank corpus is released by the Linguistic Data Consortium, catalog number LDC2012T01, `http://www.ldc.upenn.edu/` [last accessed April 8, 2014].

At EVALITA 2007 (cf. Section 3.2.2), Chronos (ITA-Chronos) was one of the participating systems. It highly outperformed the systems of all other participants with respect to both, the extraction and the normalization quality (Negri, 2007; Bartalesi Lenzi and Sprugnoli, 2007). In Section 3.6, we will compare HeidelTime's results to those of Chronos.

TERSEO – (Semi-)Automatic Approaches for Porting to other Languages

Another temporal tagger introduced in Section 3.2.5 is TERSEO. It was originally developed for Spanish and extended to process further languages, such as Italian and English. While its system architecture was already detailed, we here explain how TERSEO was extended. In contrast to Chronos, porting TERSEO to other languages was not performed manually but (semi-)automatic approaches have been tested.

First, Saquete et al. (2004) present an approach for automatic rule translation. Starting with TERSEO's Spanish patterns, online translations are performed, and expressions of the target language are linked to the normalization information of the source expressions. Then, a filtering step based on a Web search engine is applied. If expressions in the target language are not found, they are eliminated. Finally, a set of keywords in the target language is used to look for further temporal expressions, so that new rules can be learned automatically. This procedure was evaluated for the automatically developed English TERSEO system and achieved promising results on the ACE TERN 2004 test set (Negri et al., 2006).

Negri et al. (2006)[36] describe knowledge-based approaches to extend TERSEO. They tested the use of online translators, the exploitation of an annotated corpus in the target language, and the combination of both. The online translation approach is almost the same as the one of Saquete et al. (2004) for English just described above, except that two source languages (Spanish and English) have been applied, and Italian was selected as target language. Note that the English resources were automatically developed. Using two source languages is motivated by the facts that, on the one hand, the Spanish resources have been manually created, and that, on the other hand, online translations are better for English. While the detection of Italian expressions achieved good results, the correct boundary identification of the expressions and the normalization did not work well.

In the second experiment, Negri et al. (2006) used an annotated corpus of the target language (the I-CAB corpus) and translated all temporal expressions into Spanish and English. Then, the Italian expressions are assigned to appropriate normalization rules. In case of disagreement between the normalization of the Spanish and English translations, the Spanish normalization was favored since the Spanish resources were manually obtained. While this approach achieved slightly better extraction results than the translation approach, the normalization was even worse. Finally, the combined approach achieved much better results for the normalization (in particular for the most important "val" attribute). However, the results for the extraction, the correct extent detection, and the normalization are much lower than the results achieved by ITA-Chronos, i.e., by a system manually adapted to Italian (Negri et al., 2006).

Finally, Puchol-Blasco et al. (2007) suggest an approach for the multilingual extension of TERSEO based on parallel corpora. Given a sentence-aligned parallel corpus, the source language (Spanish) part of the corpus is part-of-speech tagged and processed by TERSEO, token alignment is performed on the aligned sentences of the source language and the target language, and the target language part of the corpus is part-of-speech tagged. Based on this, target language patterns can be translated from the Spanish patterns,

[36] The same procedure and results are also described by Saquete et al. (2006a).

and annotations can be added to the corpus in the target language, resulting in a temporally annotated corpus in the target language. Puchol-Blasco et al. (2007) also suggest to use the newly developed corpus as training data for machine learning approaches for the extraction of temporal expressions. However, this experiment is not performed. In addition, no evaluation results of the translated patterns or quality information of the automatically annotated corpus is provided.

In contrast, Spreyer and Frank (2008) performed a similar experiment (independent of TERSEO), namely a cross-lingual projection framework for temporal annotations, and report evaluation results. In their experiment, they used an English-German parallel corpus, automatically annotated the English corpus with a state-of-the-art system, and trained classifiers on the automatically annotated German corpus for the extraction of temporal expressions. While the annotations in the German part of the corpus are quite promising, and while the classifier achieves high precision results and can be used "as ideal starting point for a bootstrapping procedure" (Spreyer and Frank, 2008), the recall is rather low. The authors finally conclude that the approach did "not produce state-of-the-art annotations" (Spreyer and Frank, 2008).

TipSem – A Machine Learning Approach Trained on Annotated Data

A temporal tagger for English and Spanish is TipSem (Llorens et al., 2010). It was developed in the context of the TempEval-2 challenge. TipSem uses Conditional Random Fields trained on English and Spanish annotated corpora, and puts a special focus on semantic information – in particular by exploiting semantic role and semantic network information. While the same features have been used for developing the English and the Spanish models, both are trained relying on their own annotated training data. For this, the TempEval-2 training sets have been used. In addition to the extraction of temporal expressions, their classification into one of the four TimeML classes for temporal expressions (date, time, duration, and set) is performed in the same way except that the features are not used on token level but on expression level. Finally, the normalization is solved in a rule-based manner (Llorens et al., 2010).

Note that TipSem does not only extract and normalize temporal expressions but performs the full task of temporal annotation. At the TempEval-2 challenge, it performed well and achieved best results for several subtasks, in particular for Spanish temporal tagging (Verhagen et al., 2010; Llorens et al., 2010). In Section 3.6.2, we will present all systems of the TempEval-2 challenge and discuss the results. Furthermore, in Section 3.6.3, we compare HeidelTime's Spanish evaluation results with TipSem's performance when describing the results of the TempEval-3 competition.

Further Non-English Temporal Taggers

In addition to the multilingual taggers addressing Italian and/or Spanish, there is also the TimeML-compliant temporal tagger Teti for Italian only (Caselli et al., 2009). It is a rule-based system and implements WordNet-based semantic relations between temporal expressions. It is evaluated on parts of the not yet released Italian Timebank corpus (cf. Section 3.4.3). The system only performs the extraction part of temporal expressions and achieves good results for this subtask.

Another temporal tagger worth mentioning is CTEMP (Mingli et al., 2005) for temporal tagging Chinese text. It performs the tasks of extraction and normalization in a rule-based way. While they took part in the ACE TERN 2004 challenge to evaluate the extraction of temporal expressions, they had to manually annotate a corpus with normalization information to also evaluate this subtask. Compared to English temporal taggers, CTEMP achieves good results on the normalization subtask for Chinese (Mingli et al., 2005). Unfortunately, neither this corpus nor the system are publicly available.

3.4.5 Corpus Creation

In this section, we describe the corpus creation process for those of the six languages addressed in this work for which no or insufficient annotation efforts have been made so far. Then, we compare these corpora with publicly available corpora, and describe language-dependent challenges for temporal tagging, which we encountered during the integration of the corresponding languages into HeidelTime (cf. Section 3.5).

For three of the five non-English languages addressed in the context of this work, there are publicly available temporally annotated corpora, namely for Spanish, Italian, and Arabic (cf. Section 3.4.3). While the Spanish and Italian corpora contain extent and normalization information, the Arabic corpus does not contain any normalization information. In addition, there are no corpora available for the other two languages addressed in this work, German and Vietnamese. Thus, we created a German and a Vietnamese corpus and added normalization information to parts of the already existing Arabic corpus.

WikiWarsDE: a German Temporally Annotated Corpus

We developed WikiWarsDE (Strötgen and Gertz, 2011) as the German counterpart of WikiWars (Mazur and Dale, 2010), which contains parts of Wikipedia articles about important wars in history. After selecting the 22 corresponding German Wikipedia articles using Wikipedia's cross-language links,[37] we followed the developers of the WikiWars corpus for the corpus creation process. First, we manually copied sections of the Wikipedia articles describing the course of the wars and removed pictures, cross-page references, and citations. Then, all text files were converted into SGML files, the format of the ACE TERN corpora. Finally, the temporal expressions were annotated according to the TIDES TIMEX2 annotation standard. WikiWarsDE is publicly available and temporal taggers can be evaluated using the official evaluation tools of the ACE TERN evaluation.

As for the development of the Time4SMS and Time4SCI corpora, we performed a similar annotation procedure as Mazur and Dale (2010) for the creation of WikiWars. Since we already detailed this annotation procedure in Section 3.3.6 when describing the Time4SMS and Time4SCI development, we here only summarize the three-phrase annotation process: (i) Automatic pre-annotation using our own temporal tagger (HeidelTime, cf. Section 3.5), (ii) manual annotation with correcting wrong and adding missing annotations by two annotators, and (iii) manual merging and validation of the annotation of the two annotators.

To compare our inter-annotator agreement for the determination of the extents of temporal expressions to others, we calculated the same measures as the developers of the TimeBank-1.2 corpus (cf. Section 3.2.3). They calculated the average of precision and recall with one annotator's data as the key and the other's as the response. Using a subset of ten documents, they report inter-annotator agreement[38] of 96% and 83% for partial match (lenient) and exact match (strict), respectively. Our scores for lenient and exact match on the whole corpus are 96.7% and 81.3%, respectively, i.e., quite similar.

WikiWarsVN: a Vietnamese Temporally Annotated Corpus

Similar as for German, there were no temporally annotated corpora containing Vietnamese documents. Thus, we also developed a Vietnamese version of the WikiWars corpus (Strötgen et al., 2014a).

[37] Due to the shortness of the German Wikipedia article about the Punic Wars in general, we used three separate articles about the 1st, 2nd, and 3rd Punic Wars.

[38] `http://timeml.org/site/timebank/documentation-1.2.html` [last accessed April 8, 2014].

For the corpus creation, we again used the language-linked Vietnamese Wikipedia documents and manually annotated the extents of the temporal expressions and their value attributes. However, the new WikiWarsVN corpus contains only 15 articles since 7 of the 22 English articles had no linked Vietnamese version. Furthermore, the documents of WikiWarsVN are much shorter than the English and German documents and contain fewer temporal expressions due to the shortness of the Vietnamese Wikipedia articles (cf. Table 3.5, page 66). Note that in contrast to the original WikiWars corpus and WikiWarsDE, we decided to annotate temporal expressions using TimeML's TIMEX3 annotations. The main reasons for this decision were that nowadays TIMEX3 annotations are much more frequently used than TIMEX2 annotations (cf., e.g., Derczynski et al., 2012) and that we consider it more important that the annotated corpus fits to the annotation standard of our temporal tagger than that the corpus is as similar as possible to the original WikiWars corpus.

Due to the requirement of Vietnamese language skills, only one annotator performed the task of manual annotation. Although these annotations were finally discussed by this annotator and a TimeML expert without Vietnamese language skills (the author of this thesis), we consider WikiWarsVN a "silver standard" rather than a gold standard yet due to the single annotator with Vietnamese language skills and since the whole corpus had to be developed from scratch.

ACE 2005 Arabic Corpus: Adding Normalization Information

While there were no temporally annotated corpora for German and Vietnamese, there is a publicly available temporally annotated corpus for Arabic. However, the Arabic documents of the ACE 2005 training corpus do not contain normalization information, i.e., only the extents of temporal expressions are marked (cf. Section 3.4.3). To be able to evaluate a temporal tagger in a meaningful way, i.e., with respect to both, its extraction and its normalization quality, normalization information is required.

Since there were many annotation errors and missing annotations in the original corpus, it was necessary to perform a manual re-annotation instead of just adding the normalization information (value attribute) to each annotation. However, this manual re-annotation is a time-consuming task and requires Arabic language skills. Thus, in the context of the development of Arabic capabilities for our temporal tagger (Strötgen et al., 2014a), we split the rather large corpus (403 documents) into training and test sets, and only performed this re-annotation on a subset of the corpus, namely 50 randomly selected documents (the test set). Since TIMEX3 annotations are much more frequently used than TIMEX2 annotations, and because we had to perform a manual re-annotation of the extents due to the rather low quality of the original annotations anyway, we re-annotated the documents of the test set with TimeML's TIMEX3 tags.

Due to the need of language experts, only one annotator performed this re-annotation task. Similarly as for the Vietnamese corpus, the results of this re-annotation were discussed by the annotator and a TimeML expert without Arabic language skills (the author of this thesis). However, in contrast to the Vietnamese corpus, the annotation of the Arabic documents did not have to be started from scratch since the original TIMEX2 annotations had been used as basis for the full annotation.

Summary

After having presented existing temporally annotated corpora in Section 3.4.3, we presented in this section the German and Vietnamese corpora we developed in the context of this work and detailed our extensions on the Arabic part of the ACE Multilingual 2005 corpus. Table 3.5 (page 66) gives an overview of the presented non-English corpora.

3.4.6 Language Characteristics and Language-dependent Challenges

In this section, we briefly point out some language characteristics that are crucial for temporal tagging and thus result in language-dependent challenges for temporal tagging. The languages we considered in this work differ significantly with respect to several aspects. While all languages have some characteristics that have to be considered when addressing the task of temporal tagging, Arabic is probably the most difficult language we addressed in the context of this work. In Section 3.6.8, we will present an error analysis for HeidelTime. There, we will also discuss errors that occur due to language-specific challenges.

Missing Diacritics

In general, there are several challenges for Arabic natural language processing as described by Farghaly and Shaalan (2009). For instance, there are several varieties of Arabic. Fortunately, Modern Standard Arabic (MSA) is usually used in contemporary texts such as newspapers, academic papers, and modern books (Farghaly and Shaalan, 2009) so that we only consider MSA. Nevertheless, one Arabic-specific challenge is due to MSA, namely missing diacritics.

There are no letters to represent short vowels in Arabic. Formerly, diacritics represented short vowels, but in MSA these are usually not used anymore. This lack of diacritics results in many homographs, i.e., ambiguity problems. For example, without diacritics "fifth" and "five" are written as خمس (/khms/), and both "the future" and "the receiver" are written as المستقبل (/almstqbl/). However, since diacritics sometimes also occur in MSA, we had to take care of those commonly used in temporal expressions, e.g., for يومًا (/ywman/, day).

Inflection

Some languages have a more complex inflection system than others. For instance, the German, Spanish, Italian, and Arabic inflection systems are much more complex than the English one. In contrast, Vietnamese is an isolating language and there is no inflection of words (Nguyen, 1997). These language differences have to be taken into account when developing patterns to match temporal expressions.

Agreement System

While the English agreement system is rather limited, other languages have a more complex agreement system. Temporal expressions often consist of nouns and modifiers. Depending on the complexity of the agreement system, this results in several possible combinations that have to be considered. For example, there are many variations of temporal expressions in Arabic since nouns and their modifiers, e.g., adjectives, have to agree in number, gender, case, and definiteness (Farghaly and Shaalan, 2009).

Further Challenges

There are several further challenges to perform temporal tagging of documents written in the languages addressed in this work. For instance, there are two kinds of numbers in Arabic script: Western Arabic numbers (1, 2, 3, ...) and Eastern Arabic numbers (... ٣، ٢، ١). Both are used in temporal expressions.

Ambiguities also occur in many languages, for instance, in English "May" and "March" can be used as month names but also as verbs, auxiliaries or even as a noun with another meaning (as in "March of the Iron Will"). A more general example are numbers. If digits can be used in a language on their own to

refer to a year, one has to analyze whether or not a digit number refers to a year or if it is used in another context. It is much simpler if numbers are used in combination with the term "year" as it is usually done in Vietnamese, for instance. Note that sometimes, it is even not possible to solve all ambiguities as in the example "the 2000 celebrations". Here, it is not possible to decide whether "2000" refers to the year 2000 or if it is used as a regular number, in particular without deep language understanding methods.

Finally, due to the lack of inflection in Vietnamese – which is helpful to describe patterns for temporal expressions – it is sometimes difficult to decide whether underspecified expressions (e.g., "Monday") refer to the previous, current, or next Monday in a news-style document. Although such ambiguous expressions are not very frequent in Vietnamese, one should take care of tense markers such as the function words (empty words) "đã" and "sẽ" for past and future, respectively (Nguyen, 1997). These help to distinguish if one wants to refer to a time different from the "basic time" (Thompson, 1991).

3.4.7 Summary

In this section, we discussed non-English and multilingual temporal tagging. While there had been some research challenges for non-English temporal tagging, and some non-English temporally annotated corpora have been released, research on non-English temporal tagging is still limited resulting in only few publicly available temporal taggers processing more than one language in addition to English.

We also discussed some approaches on (semi-)automatically adapting a temporal tagger to further languages that have been suggested in the literature. However, in addition to the fact that most of these approaches were related to only one temporal tagger (TERSEO), the evaluation results of these approaches also showed that the quality of manually adapted temporal taggers cannot be reached – in particular for the important normalization task. While this was already the case when adapting a temporal tagger of one language to a similar one (e.g., from Spanish to Italian), it might be even more difficult to automatically address languages not related to the source language, due to the different language characteristics that should be considered by temporal taggers, as described in the previous section.

We are addressing several languages in this work. However, for some of these languages, there have not been any publicly available corpora that can be used for evaluating the extraction and the normalization of temporal expressions. Thus, we also presented the development of two new temporally annotated corpora and an extension of an existing corpus containing only extent but no normalization information so far. In addition to the fact that we can use these corpora for the evaluation of HeidelTime(cf. Section 3.6), we made further contributions to the research community by making them publicly available.

3.5 HeidelTime, a Multilingual, Cross-domain Temporal Tagger

While there are a couple of temporal taggers available as discussed in the previous sections, there is a lack of publicly available temporal taggers that can process multiple languages and documents of different domains. Although there has been significant improvements in the research area of temporal tagging in the last few years, most of the approaches still concentrate on processing English news-style documents. In the context of this work, we developed our temporal tagger HeidelTime, the first multilingual, cross-domain temporal tagger for the full task of temporal tagging, i.e., performing the extraction and the normalization of temporal expressions. We made HeidelTime publicly available and it is already used in several works by other research groups. Examples are the Computer Science Department of the University

of Illinois at Urbana Champaign (Zhao et al., 2012; Jindal and Roth, 2013), the L3S Research Center at Leibniz University Hanover (Kanhabua et al., 2012; Kanhabua and Nejdl, 2013), and the Department of Computer Science at University of Colorado (Gung and Kalita, 2012).

In this section, we present the design and implementation of HeidelTime. First, we define the system requirements we considered during the development. In Section 3.5.2, we detail HeidelTime's system architecture, which strictly separates between the source code and language-dependent resources. The language-dependent resources and HeidelTime's rule syntax are described in Section 3.5.3 and Section 3.5.4, respectively. Then, in Section 3.5.5, HeidelTime's algorithm with its domain-dependent normalization strategies is explained, and typical aspects of rule-based systems such as completeness and termination are discussed in Section 3.5.6. In Section 3.5.7, the resource development process for different languages is described before we finally present some extensions to HeidelTime and the UIMA HeidelTime kit.

A broad-coverage evaluation of HeidelTime considering multiple languages and domains will be detailed in Section 3.6, where we also compare HeidelTime's evaluation results to other temporal taggers, report on HeidelTime's processing time performance, and where we perform an error analysis, additionally. Section 3.7 outlines ongoing work and further possible improvements related to HeidelTime.

3.5.1 System Requirements

For the development of HeidelTime, we defined the following requirements:

A. **High quality**: Extraction and normalization of temporal expressions should be of high quality.

B. **Domain sensitivity**: High quality results for both tasks should be achieved on all domains described in Section 3.3, i.e., on news-, narrative-, colloquial-, and autonomic-style documents.

C. **Language extensibility**: Further languages should be easy to integrate without modifying the source code. This allows researcher not familiar with HeidelTime's programming details to develop resources for further languages.

D. **New modules**: Easy integration of modules should be possible, e.g., for further implicit expressions.

E. **Maintenance**: When needed, modifying and adding rules should be simple.

Although there are promising machine learning approaches for the extraction of temporal expressions (cf. Section 3.2.5), we developed HeidelTime as a rule-based system due to the following reasons:

- The divergence of temporal expressions is very limited compared to other named entity recognition and normalization tasks, e.g., the number of persons and organizations as well as the variety of names referring to these entities are probably infinite.
- The normalization is hardly solvable without using rules. Thus, existing approaches that rely on machine learning methods for the extraction of temporal expressions also apply rules for their normalization. In addition, it is intuitive to combine rules for the extraction and the normalization.
- Resources for additional languages can be added without an annotated corpus.
- The knowledge base can be built in a modular way, e.g., for adding events and their normalized temporal information such as "soccer world cup final 2010", which took place on July 11, 2010.

In summary, neither requirement D nor requirement E could be satisfied without using a rule-based approach. To realize the rule-based approach and to meet requirement E, we developed a well-defined rule syntax, which allows the simple modification and adding of rules. It will be detailed in Section 3.5.4.

As annotation format, HeidelTime uses the TimeML annotation standard with TIMEX3 tags for temporal expressions since it is the most recent standard. In addition, recent approaches to temporal relation extraction are based on TimeML and thus rely on TIMEX3 annotations for temporal expressions. Nevertheless, due to the similarity between TIMEX3 and TIMEX2, the tags can be converted into TIMEX2, although not all attributes are supported and we do not change the extents of temporal expressions. Similar – although in a less sophisticated way – to Saquete (2010), who used a TIMEX2 temporal tagger with a TIMEX2 to TIMEX3 transducer in the TimeML-based TempEval-2 challenge, we exploit the similarity between TIMEX2 and TIMEX3 to evaluate HeidelTime on corpora annotated according to the TIMEX2 annotation standard. Details on how to translate TIMEX2 into TIMEX3 annotations are presented by Saquete and Pustejovsky (2011) and Derczynski et al. (2012). We present our translation details in Section 3.5.8, when describing the components of the UIMA HeidelTime kit since the translations are made during the output formatting for TIMEX2-annotated corpora.

As a first official evaluation, we participated in the TempEval-2 task of extracting and normalizing English temporal expressions. HeidelTime achieved the best results for both the extraction and the normalization task (English) (Verhagen et al., 2010; Strötgen and Gertz, 2010a). In the TempEval-3 competition, HeidelTime achieved the best results for the full task of temporal tagging for both English and Spanish (UzZaman et al., 2013; Strötgen et al., 2013). Although detailed evaluation results are presented later in Section 3.6, this already verifies that requirement A and (partially) C are satisfied. How the remaining requirements B, (C,) and D are met is explained in the following sections.

3.5.2 System Architecture

A simple overview of HeidelTime's system architecture is shown in Figure 3.6. The most important feature is the strict separation between the algorithmic part, i.e., the source code, and the resources for patterns, rules, and normalization information. The resources, which are described in detail in Section 3.5.3, are organized in a modular way and read by HeidelTime's so-called resource interpreter. When new resources are added to HeidelTime, they are automatically loaded whenever they are named and built according to HeidelTime's conventions. Thus, the requirement of extensibility is satisfied (requirement D). In addition, only the resources are language-dependent. Thus, when integrating a new language, only these have to be developed or adapted from the resources of the source language, and requirement C is satisfied. Due to this feature, it is possible to develop resources for different languages without modifying the source code (cf. Section 3.5.7).

As detailed above, temporal tagging consists of the subtasks of extracting and normalizing temporal expressions. For the extraction task, HeidelTime mainly uses regular expressions that can make use of pattern resources. However, other constraints can be set as well, e.g., the part-of-speech tag of a specific token in the expression itself or before or after the temporal expression. For the normalization task, we use normalization resources containing mappings between an expression and its value in standard format. Furthermore, linguistic clues are applied to normalize ambiguous expressions. For example, the tense of a sentence may indicate the temporal relation between an expression and its reference time.

The difficulties of normalizing temporal expressions in different domains were described in Section 3.3. To allow cross-domain temporal tagging, i.e., to satisfy requirement B, HeidelTime distinguishes between

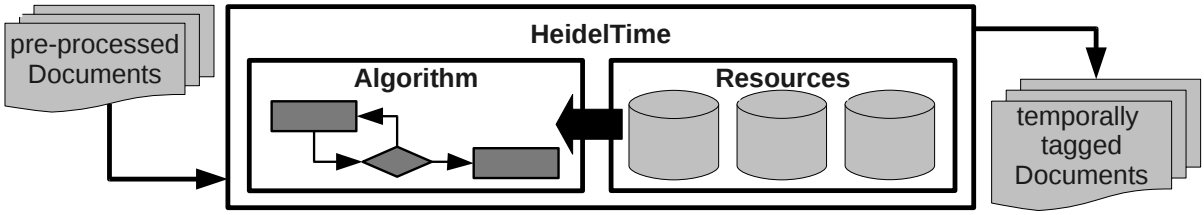


Figure 3.6: HeidelTime's simplified system architecture with algorithm (source code) and resources.

the four domains we analyzed in detail, and several of our suggested strategies to address domain-dependent challenges (cf. Section 3.3.8) are realized. For instance, domain-dependent normalization strategies for underspecified and relative expressions are developed.

In summary, HeidelTime applies different normalization strategies to documents of the following four domains: news, narrative, colloquial, and autonomic. All documents for which the document creation time is crucial and which are written in standard language are summarized as *news*. *Narratives* refer to documents for which the document creation time is usually irrelevant, i.e., the text of the document is independent of the date of writing. As for news-style documents, the document creation time is crucial for *colloquial* documents. However, the documents are not written in standard language. In documents of the *autonomic domain*, at least some of the occurring temporal expressions cannot be normalized to real points in time but only with respect to a "document-internal" (i.e., autonomic) time frame.

3.5.3 Language-dependent Resources

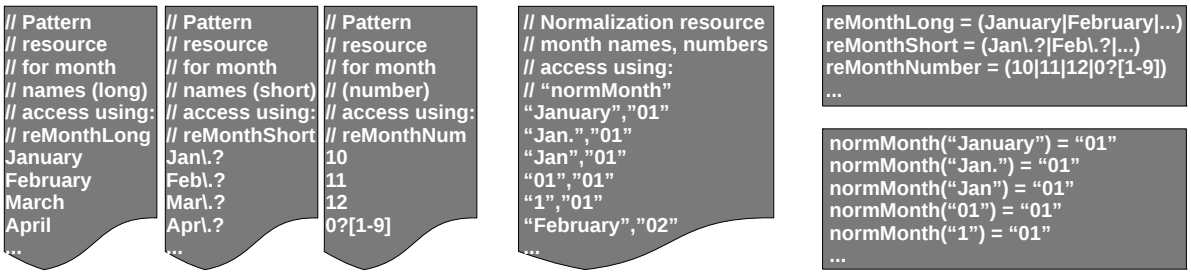
HeidelTime's resources are organized in a directory structure. For every language, three directories are used, representing the three resources (i) pattern resources, (ii) normalization resources, and (iii) rule resources. Within these directories, every resource item is represented as a file in which one can easily modify the resource or include comments and examples without influencing the resource itself.

Pattern Resources

Pattern resources are used to create regular expressions, which can be accessed by every rule. This allows to use category names instead of listing all items every time the category is needed in a rule. For example, there are patterns for month names, names of weekdays, and number words. The pattern resource files contain one disjunct per line and the regular expression is built by HeidelTime's resource interpreter when reading the resources. Figure 3.7(a) shows examples of pattern resource files, and Figure 3.7(c) (upper part) how they are translated by HeidelTime's resource interpreter.

Normalization Resources

Normalization resources contain normalized values of expressions included in the pattern resources. These values often correspond to the ISO standard for temporal information. They are used when HeidelTime assigns a value to a temporal expression, i.e., when interpreting the temporal expression. The normalization resource files are read by HeidelTime's resource interpreter, and for every file, a hash map is created. The files contain one key/value pair in each line. The key can be written in the form of a regular expression, so that the verbatim entries of the pattern resources can be used. This is particularly useful for resources of languages being rich in inflection as will be detailed in Section 3.5.7.



(a) Pattern resource files for different expressions for month names and numbers.

(b) Normalization file for month expressions.

(c) Resources as translated by HeidelTime's resource interpreter.

Figure 3.7: Pattern resource files (a) and normalization resource file (b) for expressions referring to months. Figure (c) represents how their information is represented in HeidelTime after being read by HeidelTime's resource interpreter.

Note that it is useful to split the patterns and normalization information since different rules can use different combinations of patterns which all refer to the same concept (e.g., month). Due to the single normalization resource for month information, one does not have to write similar rules multiple times but can combine patterns as disjunctions in the extraction part. An example of a normalization resource file for normalizing month expressions is given in Figure 3.7(b). How the normalization information is used in HeidelTime is shown in the lower part of Figure 3.7(c).

Rule Resources

The rule resources contain the rules for the extraction and the normalization of temporal expressions. For every type of temporal expressions (date, time, duration, and set), there is one file. In the extraction part and the normalization part of the rules, one can use the pattern resources and the normalization resources, respectively. In addition, one can define further constraints, such as specific part-of-speech tags at a specific position, or modify the extent of a temporal expression. Similar to the other resources, the rule resources are read by HeidelTime's resource interpreter and hash maps are created for the extraction, the normalization, and for all further constraints. However, due to the complexity of the rule resources, their details and the syntax of the rule language are explained separately in the following sections.

Summary

The strict separation between the source code and the resources as well as the directory structure of the resources allow the easy integration of new languages to HeidelTime. Additional modular extensions can be integrated by adding further extraction and normalization resources, e.g., for event expressions, which can be mapped to some point or interval in time. While the pattern and normalization resources can be created as described above, the rules are developed according to HeidelTime's rule syntax.

3.5.4 HeidelTime's Rule Syntax

For the extensibility and the maintenance of a rule-based system, it is important that the rules are based on and developed according to a well-defined rule syntax. In this section, we describe HeidelTime's rule syntax by detailing all the language's attributes and by presenting several examples.

rule component	description
RULENAME	contains the name of a rule
EXTRACTION	contains a regular expression-based pattern that has to be matched
NORM_VALUE	contains normalization instructions for the attribute "value"
POS_CONSTRAINT	contains one or more part-of-speech constraints that have to be satisfied
OFFSET	contains information how to change the extent of an expression extracted by the EXTRACTION part of a rule
NORM_MOD	contains normalization instructions for the attribute "mod"
NORM_QUANT	contains normalization instructions for the attribute "quant"
NORM_FREQ	contains normalization instructions for the attribute "freq"

Table 3.6: An overview of all rule components currently defined by HeidelTime's rule syntax.

Temporal Expressions as Three-Tuples

In general, HeidelTime considers every temporal expression as a three-tuple $te_i = \langle e_i, t_i, s_i \rangle$, with the expression itself (e_i), the type of the expression (t_i, with $t_i \in \{date, time, duration, set\}$), and the normalized semantic attributes of the expression (s_i). Note that s_i does not only consist of the TIMEX3 attribute *value*, but of all attributes that are subject to normalization, e.g., the *mod* attribute. However, for better readability, we start explaining HeidelTime's rule syntax with the focus on the *value* attribute.

The goal of HeidelTime is, for each temporal expression te_i in a document, to identify the expression e_i and its type t_i, and to correctly normalize its semantics s_i. To realize this goal, we developed HeidelTime's rule syntax according to which all rules have to be specified in the rule resources. While the rules are written in separate files, the source code only needs to know how to read and interpret the rules. Thus, the strict separation between the source code and the resources is supported.

Three Obligatory Components of a Rule: RULENAME, EXTRACTION, and NORM_VALUE

Table 3.6 shows all rule components currently defined by HeidelTime's rule syntax. While all components will be explained in the following using several examples, only the following three components are obligatory for every rule:

- RULENAME: Assigning a name to each rule allows to retrace which rule extracted which temporal expression. This is useful for several tasks, e.g., calculating statistics of the occurrences of different realizations of temporal expressions, and performing an error analysis and improving rules.
- EXTRACTION: Every rule contains an extraction part describing the regular expression pattern that has to be matched. In this part, one can use the pattern resources described in the previous section. In addition, one may use parentheses to group parts of the expression, which is important for the normalization of the expressions.
- NORM_VALUE: This part defines the normalized value of an expression. One can use the normalization resources described in the previous section and refer to parts of an expression using the groups defined in the extraction part of the rule. While the group()-function will be explained in more detail below, additional functions can be applied, which can be helpful to describe the normalized value of an expression. These functions are listed and explained in Table 3.7.

function	description
SUBSTRING(x,i,j)	returns a substring of string x starting at character position i and having the length j
LOWERCASE(x)	converts all characters of string x to lower case
UPPERCASE(x)	converts all characters of string x to upper case
SUM(x,y)	adds integer y to integer x

Table 3.7: Functions that can be used in the normalization parts of the rules (e.g., in NORM_VALUE) to describe the value of the normalization attributes.

The three components RULENAME, EXTRACTION, and NORM_VALUE are required and thus part of every rule. To access the pattern resources in the extraction part and the normalization resources in the NORM_VALUE part, we use the percent sign (%). For example, to access the pattern resource reMonthLong (cf. Figure 3.7(a)), one writes "%reMonthLong" in the extraction part of a rule. Accordingly, "%normMonth" can be used to access the normalization resource normMonth (cf. Figure 3.7(b)) in the normalization part of a rule. To distinguish between resources and functions in the NORM_VALUE part, function words are surrounded by percent signs, e.g., "%LOWERCASE%(x)".

An example of a simple rule, which extracts English date expressions such as "January 2, 2009" or "March 11, 1999" and normalizes their values according to the TimeML standard format (2009-01-02 and 1999-03-11 for the two examples), can be written as shown in HeidelTime's Rule Syntax Example 1. We assume, that the pattern file "reMonthLong" contains the patterns for all English names referring to months as exemplified in Figure 3.7(a), that the pattern file "reDayNumber" contains patterns for all numbers between 1 and 31, and that the pattern file "reYear4Digit" matches four digit numbers.

Rule Syntax Example 1. *A simple rule containing the three obligatory components.*

```
// matched expression: "January 2, 2009" with value "2009-01-02"
// matched expression: "March 11, 1999" with value "1999-03-11"
RULENAME="date_r1",
EXTRACTION="%reMonthLong %reDayNumber, %reYear4Digit",
NORM_VALUE="group(3)-%normMonth(group(1))-%normDay(group(2))"
```

The group()-Function

Note that in addition to parenthesis pairs, every pattern resource in the extraction part of the rule counts as one group in the group()-function. Thus, group(1), group(2), and group(3) refer to the patterns matched by "%reMonthLong", "%reDayNumber", and "%reYear4Digit", respectively. For the normalized value of the matched temporal expression, the year information can directly be used (group(3)) but the month and the day patterns have to be normalized. For this, "%normMonth()" is applied to group(1), which translates month names into their corresponding normalized values (cf. Figure 3.7(c)). In addition, "%normDay()" is applied to group(1). While two-digit numbers would not require a normalization, single-digit numbers need to be normalized, e.g., "2" has to be normalized to "02". Given the expression "January 2, 2009", replacing the groups by the matched patterns results in normalization information "2009-%normMonth(January)-%normDay(2)", which is finally resolved to "2009-01-02".

To allow for similar expressions to be matched, there is no need to write an additional similar rule but the same rule can be extended. In HeidelTime's Rule Syntax Example 2, we added the pattern for abbreviated month names (reMonthShort, cf. Figure 3.7(a)), and the pattern for ordinal numbers such as "1st" and "15th" (reDayNumberTh).

Rule Syntax Example 2. *A more complex rule still containing only the three obligatory components.*
// matched expression: "January 2, 2009" with value "2009-01-02"
// matched expression: "January 2nd, 2009" with value "2009-01-02"
// matched expression: "Mar. 11, 1999" with value "1999-03-11"
RULENAME="date_r1",
EXTRACTION="(%reMonthLong|%reMonthShort) (%reDayNumberTh|%reDayNumber), %reYear4Digit",
NORM_VALUE="group(7)-%normMonth(group(1))-%normDay(group(4))"

Based on HeidelTime's Rule Syntax Example 2, we explain the group()-function in more detail. As mentioned above, every pattern resource in the extraction part and every parentheses expression counts as a group. Thus, group(1) contains the expression matched by either "%reMonthLong" or by "%reMonthShort" while group(2) contains the expression matched by "%reMonthLong" and group(3) contains the expression matched by "%reMonthShort". In the normalization part of the rule, the month pattern has to be normalized independent of whether "%reMonthLong" or "%reMonthShort" matched successfully. Thus, we use "%normMonth(group(1))". Note that the normalization resource normMonth has to contain normalization information for long and short month names as shown in Figure 3.7(b). Similarly, group(4) contains the expression either matched by "%reDayNumberTh" with group(5) or by "%reDayNumber" with group(6), and normDay contains normalization information for all expressions in both pattern resources, e.g., that "2" and "2nd" are normalized to "02".

Further Rule Components

For some linguistic phenomena, one needs to specify further constraints to correctly extract and normalize temporal expressions. For this, we define the following attributes that can be added to a rule in addition to the rule name, extraction part, and the value normalization part. The parts of a rule norm_mod, norm_quant, and norm_freq are used to set the values of these attributes of a temporal expression in addition to the *value* attribute.

- NORM_MOD: the value of the attribute *mod* is defined here.
- NORM_QUANT: the value of the attribute *quant* is defined here.
- NORM_FREQ: the value of the attribute *freq* is defined here.
- OFFSET(group(x)-group(y)): instead of extracting the completely matched expression, the temporal expression starts with the beginning of group x and ends with the end of group y.
- POS_CONSTRAINT(group(x):y:): the part of speech tag of group x of the matched expression must be equal to y.

Examples for these rule features are described below, starting with the three further components for normalization followed by the offset and part-of-speech constraint components.

NORM_MOD, NORM_QUANT, and NORM_FREQ

To correctly normalize some expressions, in addition to the *value* attribute of a temporal expression other attributes have to be set according to the annotation standards. While all types of temporal expressions can have the modification attribute (mod), set expressions can have the quantity (quant) and frequency (freq) attributes. The parts of a rule NORM_MOD, NORM_QUANT, and NORM_FREQ are used to set the

values of these attributes of a temporal expression in addition to the *value* attribute. All the functions defined for the NORM_VALUE part as described in Table 3.7 can be used here as well.

In the extraction part of the rule date_r2 in HeidelTime's Rule Syntax Example 3, the pattern resource rePartWords is used. It contains expressions such as "the beginning of", "the end of", and "mid-", and their normalized values are defined in the ʻnormPartWords resource. This rule extracts expressions such as "mid-2002" and "the beginning of 1999" and normalizes their values to "2002" and "1999", respectively. In addition, the *mod* attribute is normalized according to the annotation standards using the normPartWords normalization resource in the NORM_MOD component of the rule. In these examples, the *mod* attributes are "MID" and "START", respectively. Note that the modifier attribute can also be used in temporal expressions of the other types, e.g., in duration expressions such as "about two weeks". Thus, the NORM_MOD part of the rule can be used in any rule with the corresponding modification pattern and normalization resources for the different types of expressions.

Rule Syntax Example 3. *A rule with the NORM_MOD component for normalizing the mod attribute, additionally.*

```
// matched expression: "mid-2002" with value "2002" and mod "MID"
// matched expression: "the beginning of 1990" with value "1990" and mod "START"
RULENAME="date_r2",
EXTRACTION="%rePartWords([ ]?)%reYear4Digit",
NORM_VALUE="group(3)",
NORM_MOD="%normPartWords(group(1))"
```

In rules for extracting and normalizing set expressions, defining the quantity and frequency attributes works analogously to the modifier attribute in this example.

The OFFSET Component

In some cases, it is necessary to define a pattern in the EXTRACTION component of a rule, which contains some context around the temporal expression itself. This is particularly useful to avoid the extraction of ambiguous expressions in the case they do not carry temporal meaning. Furthermore, it can be useful to extract context information to be able to match important information for the normalization of an expression – although the context information is not part of the extent of the temporal expression. In these cases, it is necessary to manipulate the offset of a matched pattern. For this, the OFFSET component of a rule can be used as demonstrated in the following example.

Rule Syntax Example 4. *A rule with the OFFSET component to change the extent of a temporal expression to a substring of the pattern matched by the EXTRACTION part of the rule.*

```
// matched expression: "1990-95" with offset "95" and value "1995"
RULENAME="date_r3",
EXTRACTION="%reYear4Digit(-| and )%reYear2Digit",
NORM_VALUE="%SUBSTRING%(group(1),0,2)group(3)",
OFFSET="group(3)-group(3)"
```

The rule date_r3 described in HeidelTime's Rule Syntax Example 4 matches expressions such as "1990-95" and extracts the temporal expression with the extent "95" for which the value is set to "1995". While the reYear4Digit pattern was already used in previous rules, the reYear2Digit pattern is used to match any two digit number. The normalization is done using the substring function in the norm_value part of the rule. The substring starting at position 0 with length 2 of the pattern matched by group(1) is combined with the pattern matched by group(3). In our example, the group(1) pattern is "1990", the substring is "19", and the group(3) pattern is "95". Thus the value is correctly normalized to "1995". As additional rule

component, OFFSET is used to change the offset of the expression. In our example, the offset is set to "95" – from the beginning of group(3) to the end of group(3). Without using the offset part of the rule, the whole expression "1990-95" would incorrectly be extracted as one temporal expression. Note that the expression "1990" in our example "1990-95" will be matched by another (simple) rule, which matches any reYear4Digit pattern. Thus, two temporal expressions are matched, and it is important that there are no overlapping offsets.

The POS_CONSTRAINT Component

Instead of matching patterns in the extraction part of a rule only on the lexical level, it is sometimes useful to match tokens with specific part-of-speech tags. For this, HeidelTime's rule syntax contains the rule component POS_CONSTRAINT. Using this component, one can force that a specific token is tagged with a specified part-of-speech tag by the part-of-speech tagger during the linguistic preprocessing phase. If the token is not associated to the specified part-of-speech tag, the rule will not be successful.

While this rule component can be useful for a wide range of linguistic phenomena, we will detail the POS_CONSTRAINT in HeidelTime's Rule Syntax Example 5 with a so-called negative rule.

Negative Rules

In addition to regular rules, HeidelTime also supports *negative rules*. They are used to block text fractions from being matched by other regular rules, i.e., as temporal expressions. This is useful for phrases, which look like temporal expressions, but which are used in a context in which this is not possible or unlikely.

Rule Syntax Example 5. *A negative rule to block text fractions being matched by other rules.*

```
// matched expression: "1958 miles" with value "REMOVE"
// matched expression: "2000 soldiers" with value "REMOVE"
RULENAME="date_r1_negative",
EXTRACTION="%reYear4Digit ([\w]+)",
NORM_VALUE="REMOVE",
POS_CONSTRAINT="group(2):NNS:"
```

The rule date_r1_negative in HeidelTime's Rule Syntax Example 5 is an example of such a negative rule. Assuming our rule set contains a rule that extracts temporal expressions just based on the pattern reYear4Digit, i.e., every four digit number (starting with a "1" or "2") in a text. Although such four digit numbers often refer to date expressions of the granularity year, they are often also used as numerals for count nouns. In such cases, one wants these four digit numbers to be blocked for the positive rule. This task is performed by the rule date_r1_negative. As defined in the extraction part of the rule, it extracts a four digit number followed by a token consisting of arbitrary characters. However, this arbitrary token, which is matched as group(2), must have the part-of-speech tag "NNS" as defined by the POS_CONSTRAINT part of the rule. A part-of-speech tagger assigns the "NNS" tag to plural nouns.[39] Thus, this rule extracts phrases such as "2000 soldiers" or "1958 miles". In the value normalization part of the rule, the value "REMOVE" is assigned to such expressions. The details of how the algorithm handles negative rules with "REMOVE" values, and how the matched phrases are blocked for positive rules will be described in Section 3.5.5.

[39] Note that different part-of-speech taggers use different tag sets. As will be detailed later (Section 3.5.8), we use the TreeTagger (Schmid, 1994) for part-of-speech tagging of English documents. It uses the Penn TreeBank tag set in which the NNS tag is defined to match plural nouns.

Note that the negative rule in HeidelTime's Rule Syntax Example 5 may incorrectly match expressions that refer to a year, e.g., the four digit numbers in "the 2000 celebrations" or "the 2005 treaties". However, these expressions are ambiguous and without further knowledge, solving these ambiguity problems is a tough challenge. The expressions could either refer to 2000 different celebrations and 2005 different treaties, respectively, or to the year 2000 celebrations and the treaties concluded in the year 2005. Furthermore, duration expressions such as "2000 years" are matched by the negative rules although it is clearly a temporal expression and should be extracted. How such conflicts, e.g., expressions being extracted by more than one rule, are solved will be discussed in Section 3.5.6.

Normalization of Underspecified and Relative Expressions

Using the rule syntax as described until this point and explained based on HeidelTime's Rule Syntax Examples 1 – 5, it is possible to extract and normalize temporal expressions that are explicitly mentioned in the text, i.e., for which all the information needed for the normalization is carried by each expression itself. Although only examples for date expressions have been detailed, the same methods can be used to extract and normalize time, duration, and set expressions. In addition, implicit expressions can already be extracted as well if the required resources for the normalization are available.

However, temporal information is often expressed in an underspecified and relative way (cf. Section 2.3.2). For the normalization of underspecified and relative expressions, the reference time and the relation to the reference time are important, and thus have to be considered during the normalization process. While the extraction part of rules for matching relative and underspecified expressions is similar to the ones for explicit expressions, the normalization is performed differently. For this, we set the values to expressions starting with "UNDEF" to signal that some of the normalization information is still undefined and to distinguish the values from already fully normalized ones. Depending on the domain of text that is processed (news, narratives, colloquial, or autonomic) and depending on the characteristics of the temporal expression, the reference time is determined. While the details for this normalization are explained in the next section since it is solved on an algorithmic level, the syntax for the underspecified value normalization is defined according to one of the following three formats:

- UNDEF-%normUnit(x)-REST
- UNDEF-(this|next|last)-%normUnit(x)-REST
- UNDEF-REF-%normUnit(x)-REST

The normUnit normalization resource, which is used in all three formats, contains normalized values of expressions such as day, month, and year. "REST" represents the rest of the temporal expression, which is already normalized, which might be empty, or which contains a calculation function (examples will be given below). The first format is used if the relation to the reference time is unknown, i.e., if the temporal expression is underspecified. Examples are phrases such as "In August" or "September 13" (cf. HeidelTime's Rule Syntax Example 6). Here, domain-dependent methods have to be used to identify the relation to the reference time.

Rule Syntax Example 6. *A rule matching underspecified expressions with the value being partially undefined.*

```
// matched expression: "September 13" with value "UNDEF-year-09-13"
// matched expression: "Apr. 4th" with value "UNDEF-year-04-04"
RULENAME="date_r4",
EXTRACTION="(%reMonthLong|%reMonthShort) (%reDayWordTh|%reDayNumberTh|%reDayNumber)",
NORM_VALUE="UNDEF-%normUnit(year)-%normMonth(group(1))-%normDay(group(4))"
```

The second format is used if the relation to the reference time is known, i.e., directly carried by the temporal expression. This is characteristic for relative temporal expressions. For example, "last month" refers to the previous month of the reference time. Thus, the reference time has to be identified but the relation to the reference time is determined by the expression itself. For this, the relative expression "last month" gets assigned the value "UNDEF-last-month". Other examples are expressions such as "5 days ago", which can be matched by the rule in HeidelTime's Rule Syntax Example 7. For their normalization, "REST" contains a calculation function of the form "(MINUS|PLUS)-y" with y being the amount of units that has to be added to or subtracted from the reference time.

Rule Syntax Example 7. *A rule matching relative expressions with the value being partially undefined.*
// matched expression: "5 days ago" with value "UNDEF-this-day-MINUS-5"
// matched expression: "200 years ago" with value "UNDEF-this-year-MINUS-200"
RULENAME="date_r5",
EXTRACTION="([\d]+) %reUnit ago",
NORM_VALUE="UNDEF-this-%normUnit(group(2))-MINUS-group(1)"

Finally, the third format is used to normalize expressions, for which the reference time is usually the previously mentioned temporal expression in the text – independent of the domain of the document that is processed. Examples for such expressions are "two years later" or "5 days later", which can be matched with the rule date_r6 defined in HeidelTime's Rule Syntax Example 8.

Rule Syntax Example 8. *A rule matching relative expressions with the value being partially undefined. Here, the reference time is to be identified in the text – independent of the domain of the document that is processed.*
// matched expression: "5 days later" with value "UNDEF-REF-day-PLUS-5"
// matched expression: "200 years later" with value "UNDEF-REF-year-PLUS-200"
RULENAME="date_r6",
EXTRACTION="([\d]+) %reUnit later",
NORM_VALUE="UNDEF-REF-%normUnit(group(2))-PLUS-group(1)"

All three example rules do not provide fully normalized values. Rule date_r4 matches expressions such as "September 13" and sets the value, for this example, to "UNDEF-%normUnit(year)-%normMonth(September)-%normDay(13)". After the normalization resources are resolved, this results in an underspecified value of "UNDEF-year-09-13". Rule date_r5 matches expressions like "5 days ago" and normalizes them to underspecified values. For the given example, the value is set to "UNDEF-this-%normUnit(day)-MINUS-5", which results in "UNDEF-this-day-MINUS-5". Finally, rule date_r6 matches expressions such as "5 days later". The value is normalized in an underspecified way to "UNDEF-REF-%normUnit(days)-PLUS-5", which results in "UNDEF-REF-day-PLUS-5". The final values for such expressions are then calculated internally in HeidelTime's disambiguation phase, which is domain-dependent, as will be described next.

3.5.5 HeidelTime's Algorithm with Domain-dependent Normalization Strategies

In this section, we present HeidelTime's algorithm with its different phases and the domain-dependent normalization strategies used to fully normalize underspecified and relative temporal expressions.

HeidelTime's Algorithm

As show in Figure 3.8, HeidelTime expects as input part-of-speech tagged sentences and user-specified parameters defining which types of expressions are to be annotated (parameter *annotate*) and which language and domain are used (parameters *lang* and *domain*, respectively). In an initialization phase,

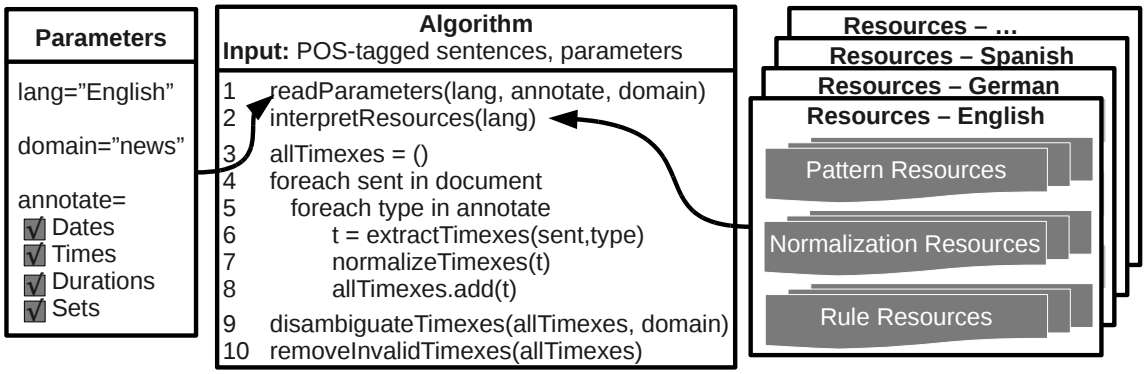


Figure 3.8: HeidelTime's algorithm reading parameters and resources.

the parameters are read (line 1) and the resources of the corresponding language are interpreted by HeidelTime's resource interpreter (line 2) as described in Section 3.5.3. Then, HeidelTime performs the extraction and normalization of temporal expressions by running the following phases: (i) the extraction phase, (ii) the normalization phase, (iii) the disambiguation phase, and (iv) the cleaning phase. In Figure 3.8, these phases are called in lines 6, 7, 9, and 10, respectively. The extraction and normalization phases are called for every sentence (line 4) and for every annotation type (line 5). Note that for each sentence, all rules are applied as will be further explained in Section 3.5.6.

Extraction Phase & Normalization Phase: Extraction and Local Normalization

During the extraction phase, the extraction parts of the rules are searched in the sentences. During the normalization phase, the – possibly underspecified – normalized values are assigned to the extracted expressions. In the previous section, we detailed the syntax of the rule language and described that further constraints (pos_constraint, offset) may have to be satisfied in the extraction phase, and further attributes (mod, freq, and quant) may have to be normalized in the normalization phase.

Disambiguation Phase: Addressing Underspecified and Overlapping Expressions

After all sentences are processed, underspecified and ambiguous expressions are subject to analysis in the disambiguation phase. For this, all extracted expressions, which are part of other temporal expressions, are removed. For example, in the phrase "On January 24, 2009, ...", HeidelTime's rules match (i) "January 24, 2009", (ii) "January 24", (iii) "January", and (iv) "2009", but all expressions except the longest one (i) are removed. If overlapping expressions are extracted, e.g., "late Monday" and "Monday morning", the situation is more difficult and thus resolved after the value normalization is finished as detailed below.

In the next step, all remaining temporal expressions are searched for values starting with "UNDEF". For these expressions, the reference time and the relation to the reference time are determined, and the values are disambiguated according to this information – depending on the domain.

Then, the overlapping expressions described above are disambiguated. For this task, different strategies may be applied. While one possible strategy, which was our first realized strategy (see, Strötgen and Gertz, 2013a), is to keep only one of two overlapping expressions, another, more promising and currently applied strategy is to merge both expressions into a single one if both expressions are of the same type

(or of types date and time) and if neither of the expressions is matched by a negative rule. In the latter case, the expression matched by the negative rule is removed. Thus, HeidelTime does not only rely on its rules but can also merge expressions similar to Chronos (Negri and Marseglia, 2004), which used specified composition rules (cf. Section 3.2.5). While determining the new extent is straightforward – e.g., "late Monday" and "Monday morning" are merged into "late Monday morning" – a distinction of cases is needed for the normalization:

- The value attribute is set in the following way: (i) If the two expressions have the same value attribute, this value is used for the merged expressions as well. (ii) If they have different value attributes, the more fine-grained value is used. (iii) If the granularities are equal for both expressions but the values are not identical, the value of the first expression is used.
- Other normalization attributes, such as the modifier attribute, are set in the following way: (i) If an attribute is identical for both expressions or only one of the overlapping expressions has an attribute, it is used for the merged expression. (ii) If two expressions have different attribute contents, the attribute content of the first expression is used.

In addition, the user is informed about overlapping expressions[40] since these indicate that the rules can probably be improved. In the example, a rule for expressions such as "late Monday morning" should be added. The user can modify the corresponding rules or create new rules, which is quite simple due to the strict separation between the source code and the resources and due to the well-defined rule syntax.

Cleaning Phase: Removing Invalid Expressions

In the cleaning phase, all invalid temporal expressions are deleted, i.e., expressions identified by negative rules and thus expressions with the value "REMOVE". Since all shorter expressions within these expressions have already been deleted in the disambiguation phase, the task of negative rules to block parts of expressions for other rules is correctly performed in the cleaning phase. The following example illustrates this procedure. Assuming the phrase "in 2000 kilometers", the expression "2000" is extracted as a temporal expression by a positive rule. However, "2000 kilometers" is matched by a negative rule (a rule similar to rule_negative_r1 presented in the Rule Syntax Example 5, page 82). During the disambiguation phase, the expression "2000" is removed since it is covered by the longer matched expression "2000 kilometers". Finally, during the cleaning phase, "2000 kilometers" is removed since it was matched by a negative rule with the value "REMOVE", so that finally no expression is matched in the phrase "in 2000 kilometers".

Domain-dependent Normalization Strategies

To further detail the disambiguation phase, we use two examples of Figure 3.2 (page 49). In the news document (Figure 3.2(a)) and the narrative document (Figure 3.2(b)), the expressions "December" and "December 25" occur. In HeidelTime's normalization phase, they are normalized to "UNDEF-year-12" and "UNDEF-year-12-25", respectively. During the disambiguation phase, these have to be fully specified. For this, HeidelTime applies domain-dependent normalization strategies (cf. Section 3.3.8). Thus, for narrative documents, HeidelTime assumes the previously mentioned temporal expression of the type date to be the reference time. Assuming a chronological order of the reference time and the underspecified expression, the value of the expression "December 25" is correctly normalized to "1979-12-25".

[40] HeidelTime outputs the following information as stderr (standard error output stream): the two overlapping expressions and the names of the rules which matched the two expressions.

For news documents, HeidelTime assumes the document creation time to be the reference time, and the relation to the document creation time has to be identified using tense information of the sentence. This is done by exploiting part-of-speech tags of the verbs in the sentence. If past tense is determined, the year of the value will be set to the year of the previous December of the document creation time. If present or future tense is identified, it will be set to the year of the December after the document creation time. In the example, the document creation time is "1998-04-28", i.e., the value of the expression "December" is correctly disambiguated to "1997-12" since the tense of the sentence (the verb "cited") is determined as past tense. In general, HeidelTime performs the domain-dependent normalization as suggested in Section 3.3.8 and as illustrated in Figure 3.5 (page 60).

Summary

In this and the previous sections, we have shown that HeidelTime's rule syntax is well-defined and can be used to extract and normalize different types and occurrences of temporal expressions. We explained some examples for English temporal expressions and detailed HeidelTime's algorithm with its domain-dependent normalization strategies. In the next sections, we will discuss typical aspects of rule-based systems, and detail how HeidelTime's resources were developed for English, but also for several other languages, and how language resources can be developed for further languages.

3.5.6 HeidelTime as a Rule-based System

In this section, we will analyze HeidelTime as a rule-based system by explaining the order of how rules are processed. Furthermore, we discuss the typical aspects of rule-based systems namely correctness, completeness, termination, confluence, consistency, and non-redundancy. First, however, we explain why it is intended that all rules are applied on each sentence.

Why all Rules Have to Be Checked

As explained in the previous section, HeidelTime's rule base may contain rules whose extraction part is completely covered by longer rules. Intuitively, one may argue that once a longer rule matches an expression, there is no need to apply shorter rules, and a logical ordering of how rules are processed could be calculated. However, there are several reasons why such a logical ordering of rules is not performed:

- The rules are applied on the sentence level since temporal expressions do not cross sentence boundaries but can be of any length within a sentence. Thus, if parts of a sentence were matched by a longer rule, all remaining parts of the sentence would still have to be processed by all other rules.
- Rules may extract overlapping expressions so that it is not useful to completely block parts of a sentence to be matched by other rules.
- In practice, HeidelTime's rules can be formulated rather complex containing several mandatory parts, which make it even more complicated to detect rules that could be left out. It is rather unlikely that the extraction parts of many rules are completely covered by other rules.
- To address the complexity of natural language, it is necessary to empower the rule developer to specify the ordering of the rules as will be further explained below.
- It may be intended by the rule developer that there are several rules extracting the same phrases as illustrated in the following example.

Assume there is a negative rule to match four-digit numbers followed by plural nouns as in HeidelTime's Rule Syntax Example 5 (page 82). In addition, assume the sentence *"The development took 2000 years"*. Clearly, despite the fact that "2000 years" would be matched by the negative rule, there will also be a positive rule to extract that phrase as a duration, e.g., by formulating the extraction part of a rule as "a number followed by a unit word such as years, days, etc.". While the negative rule is important in several other scenarios, the rule developer has to be empowered to decide that the duration rule should be considered as more important. In general, the rule developer should be able to decide which of two rules is more important if they match identical phrases. This importance of rules is covered by HeidelTime following the ordering of the rules as described next.

Order of the Rules

As just explained, it is important that the rule developer can specify which of two rules should be considered more important if they match identical phrases. Thus, we define two preference rules HeidelTime follows when processing the rules of a language:

- The importance I of a rule depends on the file in which the rule is defined. Note that there is one rule file for for each type of temporal expression. The relationship $\succ_I$ defines the importance relationship between two rules. If $r_i \succ_I r_j$, then r_i is more important than r_j.
  $I(r_i) \succ_I I(r_j)$ with r_i.file() = date; r_j.file() = time;
  $I(r_j) \succ_I I(r_k)$ with r_j.file() = time; r_k.file() = duration;
  $I(r_k) \succ_I I(r_l)$ with r_k.file() = duration; r_l.file() = set;

- Within each rule file, the importance relationship between rules is expressed by their ordering. The earlier a rule is defined, the more important is the rule.
  $I(r_i) \succ_I I(r_j)$, if r_i.offsetInFile() $< r_j$.offsetInFile() and r_i.file() = r_j.file()

Note that the importance relationship between rules is only relevant if two rules extract phrases with identical offset. In all other cases, there is no need to define the importance relationship because the rules are processed as already described above. Next, we will discuss typical aspects of rule-based systems.

Correctness and Completeness

Neither correctness nor completeness of HeidelTime's rules and resources can be guaranteed. For any language that can be processed by HeidelTime, correctness and completeness depend on the carefulness of the resource developers, the difficulty of the language, the domain, and, in general, the documents that are to be processed. In addition, HeidelTime uses linguistic preprocessing information, and if, for instance, a sentence splitter reports a sentence boundary in the middle of a temporal expression, there is no chance that HeidelTime correctly extracts and normalizes the respective expression. Thus, HeidelTime also relies on linguistic preprocessing tools.

As for several natural language processing tasks, ambiguities in natural language further avoid that correctness or completeness can be reached at all (e.g., "the 2000 celebrations", cf. Section 3.4.6, page 73). Furthermore, there are hardly any systems for any natural language processing tasks for which correctness or completeness is reached. Often, inter-annotator agreements between annotations manually created by human experts on the same documents are calculated to get an idea of an upper bound for correctness and completeness for a system (Resnik and Lin, 2010: p.276).

In the case of temporal tagging, some corpora developers reported inter-annotator agreement numbers, e.g., the developers of the TimeBank corpus reported[41] an average of precision (correctness) and a recall (completeness) of 83% for exact match and of 96% for partial match. For the normalization subtask of temporal tagging, 90% as average of precision and recall is reported for the value attribute. To answer the issues of correctness and completeness, we will describe the results of a detailed evaluation of HeidelTime on several publicly available corpora in Section 3.6.

Termination

In contrast to correctness and completeness, the termination aspect of HeidelTime can be considered fulfilled. While the extraction parts of the rules can contain expressions referring to pattern resources, these cannot be defined recursively according to HeidelTime's rule syntax. Thus, endless replacements of patterns cannot occur – neither in the initialization phase nor during processing documents.

Nevertheless, the language resources may contain syntactic errors made by the resource developers. If such syntactic errors in the rules and resources occur, these are either detected and handled by HeidelTime's resource interpreter, i.e., during HeidelTime's initialization phase, or detected later during the processing of documents. For instance, if a rule matches an expression, and the normalization component of the rule requires that a part of the matched expression is normalized using a specific normalization resource, it can occur that the respective phrase is not part of the normalization resource and thus cannot be normalized. In such cases, HeidelTime re-runs the specific sentence in debug mode and outputs information on the rule, patterns, and normalization resources being affected by the error.[42]

Confluence and Consistency

HeidelTime can be considered a confluent system. If an extraction part of a rule contains more than one reference to pattern resources, it does not matter which pattern is replaced first by HeidelTime's resource interpreter. In a similar way if a normalization component contains more than one reference to a normalization resource, it does not matter which normalization resource is replaced first. Furthermore, HeidelTime is implemented in such a way that the first occurring pattern/normalization resource is replaced first. Thus, confluence does not play an important role when analyzing HeidelTime.

Due to the fixed ordering of the rules, and since each rule is processed separately, HeidelTime can be considered as consistent. However, as for confluence, consistency is not a severe issue due to HeidelTime's system architecture.

Non-Redundancy

In contrast, HeidelTime's rules may contain redundant information. For instance, the same pattern may occur multiple times in a pattern resource, and an identical rule may be defined multiple times in a rule file. Since the order of rule processing and the importance of rules is well-defined, such redundancies may only decrease HeidelTime's processing time performance, but they do not effect correctness and completeness aspects.

[41] `http://timeml.org/site/timebank/documentation-1.2.html#iaa` [last accessed April 8, 2014].
[42] The user is informed on such issues in a similar way as for overlapping expressions (cf. Section 3.5.5, page 84).

3.5.7 Resource Development Process

In this section, we describe the resource development process for different languages. First, we detail the evolution of HeidelTime's English resources and domain capabilities. Then, we present a general strategy how to add language resources for further languages and briefly explain how the resources for the languages currently supported by HeidelTime were added. It is crucial to know which corpora have been used to develop HeidelTime's language resources to be able to interpret the evaluation results in Section 3.6, where many different corpora have been used to evaluate HeidelTime.[43]

English Resources

In the context of TempEval-2, we developed HeidelTime's first version of English resources using the TempEval-2 training data, which corresponds to the TimeBank corpus (Verhagen et al., 2010). We developed a precision- and a recall-optimized rule set (Strötgen and Gertz, 2010a), but later dropped the recall-optimized one since we decided to put HeidelTime's focus on high-quality normalization of temporal expressions rather than trying to increase the recall of the extraction task at the expense of normalization quality. Note that the TempEval-2 challenge only addressed temporal tagging of documents of the news domain, and thus, HeidelTime was developed to interpret temporal expressions according to the news domain strategy.

For processing narrative-style documents such as Wikipedia articles, we then added the second normalization strategy to HeidelTime and extended the pattern, normalization, and rule resources. However, these had only been minor extensions and the main effort was put into developing the new normalization strategy for relative and underspecified expressions (cf. Section 3.3.8 and Section 3.5.5). In addition, these adaptations were not performed using an annotated corpus since at this point in time, there has not been any temporally annotated corpus for narrative-style documents. WikiWars has been published in 2010 as the first corpus containing such documents (Mazur and Dale, 2010). Thus, in the context of our work on spatio-temporal document exploration (Strötgen and Gertz, 2010b), we manually checked the results on some Wikipedia articles, developed the narrative normalization strategy accordingly, and adapted the English resources when necessary. The result of this work corresponds to the first publicly available version of HeidelTime's English resources (initial version), and the WikiWars corpus was only used for evaluation.

After having successfully addressed the news and the narrative domain, we studied the differences to other domains, namely colloquial and scientific documents, as well as challenges and possible strategies to address them (Strötgen and Gertz, 2012b). In this context, we developed HeidelTime's normalization strategies for colloquial and autonomic documents (cf. Section 3.3.8). In addition, English-colloquial and English-scientific resources have been developed.

For the development of the English-colloquial resources, we added several non-standard language expressions, which are often used as synonyms for temporal expressions in colloquial text such as tweets

[43] Note that HeidelTime is a dynamic system and since making HeidelTime publicly available, we keep on receiving feedback with suggestions on how to improve HeidelTime. In addition, whenever we are applying HeidelTime and analyze its tagging results, we try to identify tagging errors and to think about possible improvements, which are usually easy to integrate due to HeidelTime's well-defined rule syntax. Thus, we are regularly updating HeidelTime's resources to further increase its quality for extracting and normalizing temporal expressions on different domains. Due to HeidelTime's dynamic nature, the resource development process described in this section covers the evolution of HeidelTime's resources until the current version (version 1.5, released September 17, 2013).

and short messages. For this, the entries of all pattern resources are checked for synonyms using the noslang dictionary[44] that contains more than 5,000 entries of so-called Internet slang and acronym formulations that are often used in SMS as well. Then, all synonyms are added to the pattern and normalization resources. When processing colloquial texts, one has to select "english-colloquial" as language, in addition to setting the domain to "colloquial".

For English-scientific, we added some phrases that are often used to refer to a time point zero. Furthermore, we mainly adapted the normalization resources for patterns referring to unresolvable expressions (cf. Section 3.3.8). The strategies to handle colloquial and autonomic documents, and also the pattern and normalization resources for English-colloquial and English-scientific, have been developed by analyzing the newly developed corpora Time4SMS and Time4SCI (cf. Section 3.3.6). This should be taken into account when interpreting HeidelTime's evaluation results on these corpora.

In the context of TempEval-3 (UzZaman et al., 2013), we used the TempEval-3 training data to further boost HeidelTime's extraction and normalization quality for English (Strötgen et al., 2013). However, we only used the two gold standard corpora (corrected versions of the TimeBank corpus and the Acquaint corpus, cf. Section 3.2.3) and not a newly published silver standard corpus, which contains merged results of three state-of-the-art temporal taggers. This decision was made after an initial analysis of the silver standard, which did not seem to be helpful for developing and improving a rule-based systems. The changes to improve HeidelTime's extraction and normalization quality on the TempEval-3 corpus have been validated on several other English corpora to avoid overfitting to the TempEval-3 training data.

In summary, we developed English resources for temporal tagging documents of four domains: news, narrative, colloquial, and autonomic. Note that the domain-dependent normalization strategies are language-independent, but that for autonomic and colloquial documents additional language-dependent patterns and normalization resources have been developed. In Section 3.6, we will present HeidelTime's evaluation results on different corpora and domains. Note that the evaluation corpora have not been used to develop HeidelTime's English resources until boosting HeidelTime for TempEval-3. For this, in Section 3.6, we will clearly point out whether a corpus was used during the development or exclusively as evaluation corpus.

General Resource Development Process for Further Languages

In the following, we describe the resource development process as a general strategy to add capabilities for a new language to HeidelTime. While we followed this strategy to add German, Spanish, Italian, Arabic, and Vietnamese resources, this is also the strategy we suggest to add further languages. Note that while some (semi-)automatic approaches for adapting a temporal tagger to additional languages have been described in Section 3.4.4, these did not perform as well as manually adapted systems. Thus, our strategy requires some manual effort. However, we agree with Negri (2007) that this process can be quite fast if the developer has (at least) basic knowledge about the language and is familiar with the system's architecture. Due to HeidelTime's well defined rule syntax and the strict separation between the source code and language-dependent resources, the latter point is even not very important in the case of developing HeidelTime resources for further languages.

Linguistic Preprocessing: Except for the resources, all HeidelTime internals are indeed language-independent. However, HeidelTime requires linguistic preprocessing, namely sentence splitting, tokeniza-

[44] `http://www.noslang.com/dictionary/full/` [last accessed April 8, 2014].

tion, and part-of-speech tagging (cf. Section 3.5.2). These tasks are language-dependent and have to be addressed when one wants to extend HeidelTime for further languages. As will be detailed in Section 3.5.8, HeidelTime is based on the unstructured information management architecture UIMA (Ferrucci and Lally, 2004b). Thus, the preprocessing tasks have to be performed by an analysis engine. Either one of the wrappers of the UIMA HeidelTime kit (cf. Section 3.5.8) already can process the language of interest, or an analysis engine for these preprocessing tasks has to be developed. This can usually be done by writing a UIMA wrapper for an existing linguistic preprocessing tool for the language of interest.

Resource Development Process: The linguistic compositions to form temporal expressions are language-dependent. Thus, it is important to develop language-dependent rules. However, the meaning of temporal expressions in different languages is often very similar. For example, all current HeidelTime languages contain patterns (or words) referring to names of months such as "January" (English), for which translations to the seven languages are amongst others: "Januar" (German), "januari" (Dutch), "enero" (Spanish), "gennaio" (Italian), "janvier" (French), "يناير (/ynayr/)" (Arabic), and "*tháng* một" (Vietnamese). Note that there are variations in how one refers to the month "January" in the different languages, but the meaning of "January" can be expressed by these patterns.

Translation of Pattern Files: As described in Section 3.5.3, HeidelTime's language-dependent resources contain so-called pattern files, which are read by HeidelTime's resource interpreter and later accessed by the extraction part of the rules. These pattern files contain pieces of temporal information, e.g., names of months, names of weekdays, but also numbers, which can refer to days of a month, and so on. The first step in the resource development process for a new language is to develop the pattern information. The goal is that the pattern files contain all the patterns that are usually used in the target language to form temporal expressions. For this, we start with the pattern files of the source language (usually English) and translate all the content that also exists in the target language. Note that pattern files can be removed, and new pattern files can be added if necessary.

Translation of Normalization Files: Closely related to the pattern resources are HeidelTime's normalization resources, which can be accessed by the normalization parts of the rules. Here, the meaning of the patterns is stored, for example, that "01" is the normalized value of expressions referring to the month January. It is possible to put normalization information of patterns from different pattern files into the same normalization resource. For example, there may be different patterns for expressions referring to a month which can be used in different contexts (and thus in different rules), but the normalization information of all the month patterns may be stored in the same normalization resource. Based on the source normalization resources (usually English), the normalization resources for the target language are created.

Rule Development and Iterative Resource Improvement: For the rule development, the following strategy can be applied:

1. Based on the source rules (English) and knowledge about the target language, a few simple rules for the target language are developed.
2. The training documents are processed with these simple rules and checked for incompletely matched expressions. Based on them, the simple rules can be improved and extended, and – whenever necessary – further patterns and normalization information can be added to the resources. This is, for instance, usually necessary for modifiers, which can be expressed in many different ways.

3. In the next step, the training documents are checked for undetected temporal expressions, and rules are created to match such expressions. Here, the goal should be to write the extraction part of the rules as precisely as necessary and as generally as possible. In addition, more complex source rules can be translated to achieve high coverage in the target language although such rules might not have been necessary for the training corpus of the target language. In this way, the resources for the target language can benefit from the high quality of the source language which would not be possible if a temporal tagger for the new language is developed from scratch. For instance, the Spanish HeidelTime resources benefited from the high quality of the English resources, which were used as the starting point in the Spanish development process (Strötgen et al., 2013).

4. Finally, steps (2) and (3) are applied recursively for the adapted resources. This should be done until the rules cannot be improved or modified further without worsening the already obtained results.

Note that parts of this process can be performed automatically as suggested by Negri et al. (2006) and Spreyer and Frank (2008). However, to achieve high quality temporal tagging resources for the target language, a manual inspection of the new resources is necessary to not achieve a lower quality as if a temporal tagger was tailored for the target language as reported by Negri et al. (2006).

Corpora Used during Resource Development

For developing HeidelTime resources for German (Strötgen and Gertz, 2011) as well as for Spanish, Italian, Arabic, and Vietnamese (Strötgen et al., 2014a), we followed the strategy described above. Thus, we used some corpora during the language resource development process as described in the following.

HeidelTime's German resources were developed after the English ones. For our work on multilingual document similarity (cf. Section 6.5 and Strötgen et al., 2011), we used some German Wikipedia articles to improve the German rules. However, at this point in time, we had not yet developed WikiWarsDE, and thus, we did not use the WikiWarsDE corpus for the development of the German resources. In contrast, we manually checked the Wikipedia articles for incorrectly annotated expressions to detect errors.

We then developed Spanish resources in the context of the TempEval-3 competition (Strötgen et al., 2013). Thus, we used the Spanish TempEval-3 training data for developing the Spanish HeidelTime resources. In parallel, Italian, Arabic, and Vietnamese resources were developed (Strötgen et al., 2014a). Since neither of the languages was part of the TempEval-3 challenge, we had to use other corpora during the development process. For Italian, we used the Italian TempEval-2 training corpus. For Arabic, we split the existing Arabic part of ACE multilingual 2005 training corpus into a training and test sets. Note that the training set is TIMEX2-annotated and does not contain any normalization information. Thus, special attention had to be paid to the normalization quality by manually validating the normalization of the matched expressions during the resource development. For Vietnamese, no annotated training data was available so that we used some unannotated Wikipedia articles similar as for German. In an iterative way, these were manually checked for incomplete and missed temporal expressions, as well as for the quality of the normalization of the extracted temporal expressions.

Since the normalization strategies, the rule syntax, and the English resources were already available, the development of the resources for the other languages was straightforward. Although we had to deal with some language-specific challenges (in particular for Arabic), adding HeidelTime resources for new languages is much faster than building a new temporal tagger for the language of interest.

type	attributes
Timex3	filename, sentenceId, firstTokenId, foundbyRule, timexType, timexValue, timexQuant, timexFreq, timexMod
Sentence	filename, sentenceId
Token	filename, sentenceId, tokenId, pos
DCT	filename, value, timexId
Timex3Interval	Timex3, timexValueEB, timexValueLB, timexValueEE, timexValueLE

Table 3.8: All types and their attributes as defined in HeidelTime's UIMA type system.

Motivated by the simplicity of adding language resources to HeidelTime, the resources for the other two languages currently supported by HeidelTime (Dutch and French) have been independently developed by other researchers: van de Camp and Christiansen from Tilburg University addressed Dutch (van de Camp and Christiansen, 2012) while Moriceau and Tannier from LIMSI (Paris) addressed French (Moriceau and Tannier, 2014). Both followed a similar strategy as we did for the other languages.

3.5.8 The UIMA HeidelTime Kit

In this section, we will present the UIMA HeidelTime kit containing several collection readers, analysis engines, and CAS consumers (cf. Section 2.5). Furthermore, we describe the UIMA HeidelTime type system containing all UIMA types that are needed for processing documents with HeidelTime within a UIMA pipeline.

The HeidelTime Type System

An overview of the types defined in the HeidelTime type system is given in Table 3.8 together with the attributes of each type. In addition to the listed attributes, every type has the native UIMA attributes *begin* and *end*, which are used to set the offset of an annotation in the *documentText*.

The types *Sentence* and *Token* are usually annotated during preprocessing by the analysis engines performing sentence splitting and tokenization. While sentence annotations contain only extent, filename, and id information, token annotations have the *pos* attribute additionally, in which the part-of-speech annotation is stored. This information is usually added to existing token annotations by an analysis engine performing part-of-speech tagging. The *DCT* type contains the document creation time of a document and is usually set by a collection reader when accessing a document. The *Timex3* type is used to annotate temporal expressions with the attributes representing several of the TimeML's TIMEX3 attributes. Finally, the *Timex3Interval* type is used by our HeidelTime extension to annotate temporal expressions and combinations of temporal expressions as intervals with earliest and latest begin (EB and LB) and end points (EE and LE). Examples for such intervals will be given below.

UIMA Collection Readers

The HeidelTime kit (version 1.5) contains three collection readers for accessing and preparing input documents of all kinds of temporally annotated corpora.

- *TempEval-2 Reader*: This component reads the TempEval-2 annotated data sets and creates a CAS object for each document. It sets the documentText variable for each CAS object as well as the document creation time. In addition, sentence and token information is directly annotated since they are provided in the TempEval-2 data sets.
- *TempEval-3 Reader*: This reader was developed in the context of the TempEval-3 contest to access the TempEval-3 corpora. It also sets the documentText variable and the document creation time for each document. In contrast to the TempEval-2 data, the TempEval-3 data does not contain any token or sentence information so that no further annotations are added.
- *ACE Tern Reader*: This collection reader can be used to access all corpora formatted according to the ACE format, e.g., the ACE TERN 2004 corpus and WikiWars. Similar to the other two readers, it annotates the documentText and the document creation time for each document.

UIMA Analysis Engines

The UIMA HeidelTime kit does not only contain HeidelTime but also wrappers for tools performing linguistic preprocessing in several languages. In addition, an interval tagger is included as extension to HeidelTime, which may be useful in several scenarios although its output is not according to TimeML.

- *HeidelTime*: The HeidelTime analysis engine performs the temporal tagging task as described in this chapter and annotates all extracted and normalized expressions using the Timex3 type.
- *TreeTagger Wrapper*: This analysis engine wraps the TreeTagger (Schmid, 1994) so that it can be used within UIMA. In addition to the part-of-speech tagging task, we also use the wrapper for sentence splitting and tokenization. For HeidelTime's current version, we use the TreeTagger for preprocessing the following languages: English, German, Dutch, Spanish, French, and Italian.
- *Stanford POS Tagger Wrapper*: Similar to the TreeTagger wrapper, this analysis engine wraps the Stanford part-of-speech tagger (Toutanova et al., 2003) and annotates documents with sentence, token, and part-of-speech information. While the main motivation to include a wrapper for the Stanford tagger was to perform Arabic preprocessing, the Stanford tagger can also be used to perform part-of-speech tagging of further languages, e.g., English and German.
- *JVnTextPro Wrapper*: Since neither the TreeTagger nor the Stanford POS tagger contain capacities for processing Vietnamese text, we included a wrapper for JVnTextPro, a tool to process Vietnamese text (Nguyen et al., 2010). This analysis engine annotates sentence, token, and part-of-speech information in Vietnamese documents.
- *Interval Tagger*: This analysis engine can be used as add-on to HeidelTime. For each temporal expression of the type *date* and *time* it creates interval annotations containing the earliest and latest start and end points of the interval. In addition, one can define rules to match interval expressions being built of two regular TIMEX3 expressions. For example, while HeidelTime annotates in the phrase "From July to November 2012", the two temporal expressions "July" and "November 2012" with the value attribute being set to "2012-07" and "2012-11", respectively, the Interval Tagger matches the whole expressions and annotates the earliest begin value as "2012-07-01", the latest begin value as "2012-07-31", the earliest end value as "2012-11-01", and the latest end value as "2012-11-30". Note that such annotations do not follow TimeML specifications but they can be useful for several tasks relying on temporal information.

- *Annotation Translator*: Finally, the annotation translator analysis engine can be used to map annotations of one type system into annotations of another type system. For example, if a user applies its own sentence splitter, tokenizer, and part-of-speech tagger to create sentence and token annotations of a specific type system, the Annotation Translator can be adapted to translate these annotations into sentence and token annotations as defined in the HeidelTime type system. Note, however, that if another part-of-speech tagger is used, its tag-set should be identical to the one originally used by HeidelTime. Otherwise, rules relying on part-of-speech information may not work as expected or need to be adapted.

UIMA CAS Consumer

The three CAS consumers in the UIMA HeidelTime kit are mainly developed to format the UIMA output in such a way as it is needed to evaluate HeidelTime on several gold standard corpora.

- *TempEval-2 Writer*: This CAS consumer should be used in combination with the TempEval-2 collection reader. The output of the UIMA pipeline is formatted in such a way that it is possible to directly run the official TempEval-2 evaluation scripts.
- *TempEval-3 Writer*: Similar to the TempEval-2 Writer, the TempEval-3 Writer outputs temporally annotated documents in such a way as required by the official TempEval-3 evaluation scripts. However, due to the similarity between the format required by the evaluation scripts and the TimeML document format, it can also be used for creating standard output, i.e., if the goal is not to evaluate HeidelTime on TempEval-3 data sets.
- *ACE Tern Writer*: Finally, the ACE Tern Writer outputs documents with temporal expressions annotated with TIMEX2 tags. Thus, it is possible to perform evaluations on corpora formatted in the ACE style. Since these corpora are annotated according to TIMEX2 annotation guidelines, it is necessary that HeidelTime's TIMEX3 annotations are translated accordingly. Note that besides the differences between TIMEX2 and TIMEX3, we do not change the extent of temporal expressions. However, for set expressions, we adapt the value attribute and add the set attribute according to the TIMEX2 annotation guidelines. All attributes except the value attribute are not adapted at all since our evaluations focus on the value attribute for the normalization task. However, a more sophisticated translation of TIMEX3 to TIMEX2 annotations would probably result in better HeidelTime evaluation results on TIMEX2-annotated corpora.

Availability

In addition to the UIMA HeidelTime kit, we also made available a Java standalone version, which can be used outside of a UIMA pipeline. Both versions are continuously maintained and are important contributions to the research community.

Using the UIMA HeidelTime kit in combination with our evaluation script package containing the official ACE, TempEval-2, and TempEval-3 evaluation scripts as well as several further scripts (e.g., for corpus preparation), all evaluation results presented in the next section can be reproduced.[45]

[45] HeidelTime's evaluation numbers as well as instructions how to reproduce the evaluation results can be found at `http://code.google.com/p/heideltime/` [last accessed April 8, 2014].

3.6 HeidelTime's Evaluation Results

During the development of HeidelTime, we performed a wide range of evaluations. After describing evaluation measures and settings, we present the outcome of our participations in TempEval-2 (Section 3.6.2) and TempEval-3 (Section 3.6.3). Here, we compare HeidelTime's evaluation results to those of the systems of the other participants. When available, we list evaluation details of other temporal taggers in the following sections as well for further comparisons between HeidelTime and other state-of-the-art systems.

In Section 3.6.4, further English corpora which are from different domains are subject of analysis. The value of HeidelTime's domain-sensitive temporal tagging approach is explicitly demonstrated in Section 3.6.5 by presenting our cross-domain evaluation experiment. After describing evaluations on non-English corpora (Section 3.6.6), we present a time performance analysis (Section 3.6.7) and finalize the section by discussing the findings of a multilingual error analysis (Section 3.6.8).

3.6.1 Evaluation Measures

When reporting evaluation results of temporal taggers, two tasks are to be considered: the extraction of temporal expressions and their normalization. Both tasks can be evaluated with the widely used measures of precision (P), recall (R), and f_1-score (F1) (cf. Section 2.6). These measures have also been used in the research competitions described in Section 3.2.2 to evaluate the participants' systems.

The extraction quality of temporal expressions are usually evaluated on an expression level. Thus, one can distinguish between strict matches (e.g., gold annotation "Monday morning" versus system annotation "Monday morning") and relaxed matches (e.g., gold annotation "Monday morning" versus system annotation "Monday"). In the TempEval-2 challenge, the evaluation was performed on a token level, i.e., each token was evaluated separately. Thus, we will present the TempEval-2 evaluation results using the token-level performance in the next section to show the official evaluation results of HeidelTime and the other participants' systems. However, evaluating temporal taggers on the token level was only applied in the TempEval-2 challenge. The TempEval-3 evaluation, as most other evaluations of temporal taggers in general, is performed on the expression level. Thus, all evaluation results except the official TempEval-2 results are also based on the expression level – a much more intuitive procedure.

For the normalization, we also follow the TempEval-3 evaluation style and consider the "value" attribute as most important. However, there are again different possibilities to calculate the normalization quality of a temporal tagger: It can be evaluated either with respect to all expressions in the gold standard or to all expressions correctly identified by the system. While the second method is used by the ACE TERN and the TempEval-2 scripts, we argue similar to Ahn et al. (2005) and the TempEval-3 organizers (UzZaman et al., 2013) that the first one is more meaningful. For the sake of completeness, we give the following evaluation results of HeidelTime on all corpora:

- *relaxed*: relaxed extraction
- *strict*: strict extraction
- *value*: correct value normalization, based on correctly extracted expressions only
- *relaxed+value*: relaxed extraction with correct value normalization
- *strict+value*: strict extraction with correct value normalization

	extraction			normalization		
	P	R	F1	value	type	extraction method
HeidelTime-1	90	82	**86**	**85**	96	rule-based
HeidelTime-2	82	91	**86**	77	92	rule-based
TRIOS	85	85	85	76	94	CRF + rule-based filtering
TRIPS	85	85	85	76	94	CRF + rule-based filtering
TipSem	92	80	85	65	92	CRF
KUL Run 2	85	84	84	55	91	maximum entropy classifier
KUL Run 3	85	84	84	55	91	maximum entropy classifier
Edinburgh-LTG	85	82	84	63	84	rule-based
USFD2	84	79	82	17	90	rule-based
KUL	78	82	80	55	91	maximum entropy classifier
KUL Run 5	75	85	80	55	91	maximum entropy classifier
KUL Run 4	76	83	80	51	91	maximum entropy classifier
TipSem-B	88	60	71	59	88	CRF
TERSEO	76	66	71	65	**98**	rule-based
JU-CSE	55	17	26	00	00	rule-based
HeidelTime 1.5	87.3	86.0	**86.7**	**86.0**	96.0	rule-based

Table 3.9: Results of the TempEval-2 temporal tagging task for English (Verhagen et al., 2010), HeidelTime's current performance (version 1.5), and the extraction methods of all systems.

For most NLP and IR tasks relying on temporal information, it is important that temporal expressions are normalized correctly while it is rather less important if the expressions are matched partially or completely. Thus, we argue that the results for relaxed matching with correct value normalization are most meaningful (relaxed+value). The *relaxed+value* f_1*-score* has also been the official ranking measure for the full task of temporal tagging in the TempEval-3 challenge (UzZaman et al., 2013).

3.6.2 HeidelTime at TempEval-2 (English)

At the TempEval-2 competition, we participated in the temporal tagging task for English documents (Strötgen and Gertz, 2010a). Eight teams addressed this task with a total number of 15 runs (Verhagen et al., 2010). We submitted two HeidelTime runs – one with a precision-optimized rule set (HeidelTime-1) and one with a recall-optimized rule set (HeidelTime-2).

In Table 3.9, the official TempEval-2 evaluation results are shown. For the extraction, the measures precision, recall, and f_1-score are calculated on a token level. For the normalization, the accuracies of the two TIMEX3 attributes type and value are shown. Note that they are calculated according to all expressions correctly identified by the corresponding system and not according to all expressions in the gold standard (cf. Section 2.6.1).

Both HeidelTime runs outperformed all other systems with respect to both, the extraction and the normalization quality. While the extraction results of several systems are quite similar (12 of 15 runs by 6 of the 8 participating teams reached an f_1-score equal to or above 80%), HeidelTime achieved the best results with an f_1-score of 86% with both runs. In addition, the recall-optimized rule set achieved the best recall of all systems (91%). In contrast to the extraction results, there are large differences in the

normalization quality of the systems. Although the results are more difficult to compare since the systems' recall measures should be considered when interpreting the type and value accuracies, HeidelTime clearly outperforms all other systems. With 77% value accuracy, even HeidelTime's recall-optimized run achieves better value normalization quality than all other systems.

While all systems in the TempEval-2 challenge used rule-based approaches for the normalization task, the approaches for the extraction of temporal expressions differed. Thus, in addition to the evaluation results, we also list the methods used by the different systems for the extraction subtask in Table 3.9. On the one hand, TRIPS/TRIOS (UzZaman and Allen, 2010) and TipSem (Llorens et al., 2010) used conditional random fields for the extraction and KUL's approach is based on a maximum entropy classifier. On the other hand, Edinburgh-LTG (Grover et al., 2010), USFD2 (Derczynski and Gaizauskas, 2010), and HeidelTime are all rule-based. These six approaches all achieved competitive extraction results. The two remaining systems, TERSEO (Saquete, 2010) and JU-CSE (Kumar Kolya et al., 2010), are also rule-based. TERSEO, which usually creates TIMEX2 annotations and which is thus used in combination with a TIMEX2 to TIMEX3 (T2T3) transducer, is "a knowledge-based system for Spanish automatically extended to English" (Saquete, 2010). This automated process is probably the main reason for its low recall. JU-CSE's rule set has been at a very initial state according to the developers (Kumar Kolya et al., 2010).

Thus, the main findings of the results of the TempEval-2 temporal tagging task are that the extraction part can be successfully addressed by rule-based and machine learning approaches. In contrast, the normalization task was addressed by all systems in a rule-based manner. The latter fact is one of the main motivations for developing rule-based approaches for temporal tagging as stated by Grover et al. (2010) in accordance with our opinion: "The main motivation for [a rule-based approach] arises from the need to ground (provide temporal values for) [...] [temporal expressions] and the rules for the grounding are most naturally implemented as an elaboration of the rules for recognition" (Grover et al., 2010).

HeidelTime's current version 1.5 achieves slightly better results than the two submitted HeidelTime runs from 2010. While we dropped the recall-optimized approach, we were able to slightly further improve the recall with only minor decrease of the precision and without decreasing the normalization quality.

3.6.3 HeidelTime at TempEval-3 (English and Spanish)

TempEval-3 is the follow-up competition of TempEval-2. We addressed the task of temporal tagging by tuning HeidelTime's English resources and developing new Spanish resources (Strötgen et al., 2013).

English

Nine teams submitted 20 unique runs for the English temporal tagging task (UzZaman et al., 2013). Table 3.10 shows the results ranked by the official TempEval-3 ranking measure *value F1*. In addition to the results of the participants, HeidelTime's current performance is shown as well as the results of TIPSem, which was developed by one of the TempEval-3 organizers in the context of TempEval-2 (cf. Section 3.4.4, Llorens et al., 2010). Precision, recall, and f_1-score are given for strict and relaxed matching. For the normalization, the value F1 and type F1 measures are provided. Additionally, the extraction methods used by the different approaches are shown to allow for a meaningful analysis of the evaluation results.

The characteristics of our three runs are as follows: HeidelTime 1.2 is the HeidelTime version which was available when the TempEval-3 experiments took place. HeidelTime-bf is a bug-fixed version that was

	strict extraction			relaxed extraction			normalization		
	P	R	F1	P	R	F1	value F1	type F1	extraction method
HeidelTime-t	83.85	78.99	81.34	93.08	87.68	90.30	**77.61**	82.09	rule-based
HeidelTime-bf	80.77	76.09	78.36	90.00	84.78	87.31	72.39	79.10	rule-based
HeidelTime-1.2	80.15	76.09	78.07	89.31	84.78	86.99	72.12	78.81	rule-based
NavyTime-1,2	78.72	80.43	79.57	89.36	91.30	**90.32**	70.97	80.29	rule-based
ManTIME-4	78.86	70.29	74.33	95.12	84.78	89.66	68.97	77.39	CRF, post proc.
ManTIME-6	81.98	65.94	73.09	98.20	78.99	87.55	68.27	79.52	CRF, post proc.
ManTIME-3	76.07	64.49	69.80	94.87	80.43	87.06	67.45	76.08	CRF
SUTime	78.72	80.43	79.57	89.36	91.30	**90.32**	67.38	80.29	rule-based
ManTIME-1	78.57	63.77	70.40	97.32	78.99	87.20	67.20	77.60	CRF
ManTIME-5	77.68	63.04	69.60	97.32	78.99	87.20	67.20	77.60	CRF
ManTIME-2	79.82	65.94	72.22	97.37	80.43	88.10	66.67	76.98	CRF, post proc.
ATT-2	90.57	69.57	78.69	98.11	75.36	85.25	65.57	77.87	MaxEnt
ATT-1	91.43	69.57	79.01	99.05	75.36	85.60	65.02	78.19	MaxEnt
cleartk-1,2	85.94	79.71	**82.71**	93.75	86.96	90.23	64.66	**84.21**	SVM,Logit
JU-CSE	81.51	70.29	75.49	93.28	80.43	86.38	63.81	75.49	CRF
KUL-1,2	76.99	63.04	69.32	92.92	76.09	83.67	62.95	74.10	Logit, post proc.
KUL-ABC	81.42	66.67	73.31	92.04	75.36	82.87	62.15	73.31	Logit, post proc.
cleartk-3,4	83.19	71.74	77.04	94.96	81.88	87.94	61.48	81.71	SVM, Logit
ATT-3	87.63	61.59	72.34	97.94	68.84	80.85	60.43	75.74	MaxEnt
FSS-TimEx	52.03	46.38	49.04	90.24	80.43	85.06	58.24	68.97	rule-based
TIPSem (TE2)	93.46	72.46	81.63	97.20	75.36	84.90	65.31	75.92	CRF
HeidelTime 1.5	83.85	78.99	81.34	93.08	87.68	90.30	**77.61**	82.09	rule-based

Table 3.10: Results of the TempEval-3 temporal tagging task for English (UzZaman et al., 2013), HeidelTime's current performance (version 1.5), and the extraction methods of all systems.

never officially released but which contains several improvements and bug fixes developed independently from TempEval-3. Finally, HeidelTime-t is the HeidelTime version which was tuned in the context of TempEval-3. For this, we used the gold standard training data provided by the organizers. In addition to language-independent changes, e.g., century and decade expressions are now normalized strictly following the TimeML annotation guidelines, we also improved the English rules and resources based on observations in the training data. Examples are (i) more negative rules to better avoid the extraction of ambiguous expressions (e.g., may, march, fall) if they do not refer to a date, and (ii) more combinations of articles and modifiers were included to several rules. However, note that HeidelTime was already a state-of-the-art tool for English temporal tagging so that the changes were rather minor.

As shown in Table 3.10, all three HeidelTime runs outperformed all other systems for the full task of temporal tagging represented by the *value F1* measure. In addition, by following the TimeML annotation guidelines more closely with the tuned HeidelTime version (HeidelTime-t), we were able to further improve HeidelTime's performance. Similar to the systems of the TempEval-2 challenge, the normalization of temporal expressions was addressed by all systems using rule-based approaches while there had been several different approaches to address the extraction task. Furthermore, the extraction quality of several systems are very close so that the organizers concluded that "rule engineering and machine learning are equally good at timex recognition" (UzZaman et al., 2013).

	strict extraction			relaxed extraction			normalization		
	P	R	F1	P	R	F1	value F1	type F1	method
HeidelTime	90.91	80.40	**85.33**	96.02	84.92	**90.13**	**85.33**	**87.47**	rule-based
FSS-TimEx	65.83	39.70	49.53	86.67	52.26	65.20	50.78	62.70	rule-based
TIPSemB-F (TE2)	88.51	77.39	82.57	93.68	81.91	87.40	71.85	82.04	CRF
HeidelTime 1.5	90.91	80.40	**85.33**	96.02	84.92	**90.13**	**85.33**	**87.47**	rule-based

Table 3.11: Results of the TempEval-3 temporal tagging task for Spanish (UzZaman et al., 2013), HeidelTime's current performance (version 1.5), and the extraction methods of all systems.

The best results for relaxed extraction achieved SUTime (Chang and Manning, 2013) and NavyTime (Chambers, 2013), which uses SUTime for the extraction task and only contains improvements for the normalization task. With HeidelTime's f_1-score being only 0.02 percentage points below SUTime, two rule-based systems achieved the best extraction performance for relaxed matching. The best system for strict extraction of temporal expressions is clearTK (Bethard, 2013b) using machine learning methods for the extraction. For the normalization, the clearTK system relied on the TIMEX3 normalization tool TimeN (Llorens et al., 2012a: cf. Section 3.2.5) but could not achieve as good results as HeidelTime. Although HeidelTime's current version slightly differs from the version used in the TempEval-3 contest, these changes did not influence the evaluation results on the TempEval-3 evaluation corpus.

Spanish

As already pointed out in Section 3.2.2, the TempEval-3 temporal tagging task is also an evidence that the main focus of research on temporal tagging is on processing English documents. Only two of the nine teams addressed the Spanish temporal tagging task: HeidelTime and FSS-TimEx (Zavarella and Tanev, 2013). In Table 3.11, the official TempEval-3 temporal tagging results for Spanish are shown for the two systems and, additionally, for TipSemB-F, the winner of the TempEval-2 Spanish temporal tagging task (cf. Section 3.4.4, Llorens et al., 2010).

HeidelTime does not only achieve better results than FSS-TimEx but also outperforms TipSemB-F in particular with respect to the *value F1* measure. Thus, HeidelTime can be considered as new state-of-the-art system for temporal tagging Spanish documents. In addition, the TempEval-3 evaluation results demonstrate that high quality HeidelTime resources can be developed for a new language without modifying the source code (cf. Section 3.5.3 and Section 3.5.7).

As described in Section 3.5.7, we developed the Spanish HeidelTime resources using the English resources as a starting point. In addition, we used the Spanish TempEval-3 training set for improving the development of the patterns, normalization information, and rules. When analyzing HeidelTime's evaluation results, we recognized that the Spanish resources highly benefited from using the English resources as basis for the development because HeidelTime's Spanish resources cover much more diverse expressions than available in the Spanish training data. Thus, the strategy of using high quality HeidelTime resources of one language (English) as starting point for developing resources for another language turned out to be a very successful approach (Strötgen et al., 2013).

3.6.4 Further Results on English Corpora

In this section, we present further evaluation results on English corpora and compare our results with other systems if such results are available. Details about the corpora and some of the systems were described in Section 3.2.3 and Section 3.2.5, respectively. HeidelTime's evaluation results (version 1.5) and the results of the other systems are presented in Table 3.12.

Results on English News and News-style Corpora

In Table 3.12(a) and Table 3.12(b), the results on the ACE TERN 2004 and 2005 training corpora are presented, respectively. Note that the corpora are TIMEX2-annotated and that we did not use these corpora to develop HeidelTime's English resources. On the ACE TERN 2004 training corpus, HeidelTime achieves better results than GUTime (Mani and Wilson, 2000a) for which only f_1-scores are published. The developers of DANTE published the results of their system on the ACE TERN 2005 training corpus using two rule sets, an initial one and a rule set, which was improved using the corpus itself (Mazur and Dale, 2010). HeidelTime achieves much better results than DANTE's initial rule set and only slightly worse results than DANTE's improved rule set with respect to normalization quality, although DANTE is a TIMEX2-compliant temporal tagger and the corpus was used for improving DANTE's rule set. The latter fact also explains DANTE's very high numbers for the extraction task.

In Table 3.12(c) and Table 3.12(d), evaluation results on the TimeBank corpus are shown (1.2 (c) and TempEval-3 (d) versions of TimeBank). Note that we developed the first version of HeidelTime's English resources in the context of the TempEval-2 challenge where the training data contains the TimeBank corpus. In addition, the TempEval-3 TimeBank version was used to tune HeidelTime's resources in the context of the TempEval-3 challenge. Thus, HeidelTime's results on the TimeBank corpus are not results on unseen data. As shown in Table 3.12(c), three further taggers were evaluated on the TimeBank-1.2 corpus: a rule-based system (Boguraev and Ando, 2005), a machine learning approach using a maximum entropy classifier (Kolomiyets and Moens, 2009), and a hybrid approach using conditional random fields and rule-based filtering (UzZaman and Allen, 2011).[46] The authors of all three approaches only provide evaluation results for the extraction of temporal expressions. HeidelTime achieves better results than the other three taggers. Furthermore, we present results for the normalization task demonstrating the high quality of HeidelTime's English resources. Reasons why we did not achieve even better results since we used the corpus for tuning HeidelTime's resources will be discussed in Section 3.6.8.

In addition to the TimeBank corpus, the TempEval-3 organizers provided a cleaned-up version of the AQUAINT corpus as gold standard training data. Our evaluation results on this corpus are shown in Table 3.12(e). Finally, for the sake of completeness, we provide more detailed evaluation results on the TempEval-3 platinum corpus, which we used for evaluation purposes only (Table 3.12(f)).

Results on English Non-News Corpora

In addition to the news-style corpora, we evaluated HeidelTime on corpora of other domains. We used the WikiWars corpus (Mazur and Dale, 2010) for narrative-style documents, and our newly developed corpora Time4SMS and Time4SCI (Strötgen and Gertz, 2012b) for colloquial and autonomic documents, respectively (cf. Section 3.3.6). In Table 3.13, HeidelTime's evaluation results are presented.

[46] The second and the third approaches were also evaluated in the context of the TempEval-2 challenge (Kolomiyets and Moens, 2010; UzZaman and Allen, 2010); cf. Section 3.6.2.

(a) ACE TERN 2004 training corpus.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	95.0	78.7	86.1	86.7	71.8	78.5	87.0	87.5	87.3	82.7	68.5	74.9	77.9	64.5	70.6
GUTime			85			78			82						

GUTime: `http://timeml.org/site/tarsqi/modules/gutime/index.html` [last accessed April 8, 2014].

(b) ACE TERN 2005 training corpus.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	88.5	75.2	81.3	75.5	64.1	69.3	73.7	76.3	75.0	65.2	55.4	59.9	61.3	52.1	56.4
DANTE (init.)	71	87	78	53	65	58				34	42	37	30	36	33
DANTE (imp.)	88	93	90	75	79	77				63	67	65	57	60	58

DANTE (initial and improved rule sets): Mazur and Dale (2010).

(c) TimeBank (version 1.2).

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	91.8	91.6	91.7	85.8	85.6	85.7	87.4	87.4	87.4	80.2	80.1	80.1	76.3	76.2	76.2
BA-05	85.2	95.2	89.6	77.6	86.1	81.7									
KM-09	87.2	83.6	85.2	86.6	79.6	82.8									
UA-11	95.4	86.5	90.7	86.5	78.5	82.3									

BA-05: Boguraev and Ando (2005), with relaxed as identical right boundaries instead of overlap.
KM-09: Kolomiyets and Moens (2009), 10-fold-cross validation on the corpus.
UA-11: UzZaman and Allen (2011), 10-fold-cross validation on the corpus.

(d) TimeBank (TempEval-3 version).

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	92.2	90.8	91.5	85.4	84.2	84.8	86.4	86.4	86.4	79.6	78.4	79.0	73.7	72.7	73.2

(e) AQUAINT (TempEval-3 version).

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	90.8	92.6	91.7	80.2	81.7	80.9	79.5	79.5	79.5	72.2	73.6	72.9	63.7	64.9	64.3

(f) TempEval-3 Platinum English evaluation corpus.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	93.1	87.7	90.3	83.8	79.0	81.3	85.9	85.9	85.9	80.0	75.4	77.6	72.1	67.9	69.9

Table 3.12: English evaluation results on publicly available news-style corpora.

(a) WikiWars.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	95.6	82.0	88.3	87.5	75.1	80.8	88.6	89.2	88.9	84.7	72.7	78.2	80.2	68.8	74.1
DANTE (init.)	90	75	82	42	35	38				22	18	20	19	16	17
DANTE (imp.)	98	99	99	95	95	95				59	60	59	58	59	58

DANTE (initial and improved rule sets): Mazur and Dale (2010).

(b) Time4SMS.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	99.4	91.3	95.2	98.2	90.2	94.1	97.1	97.1	97.1	96.5	88.7	92.4	96.1	88.3	92.1

(c) Time4SCI.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	95.9	65.6	77.9	90.0	61.6	73.1	89.5	89.5	89.5	85.8	58.8	69.8	80.4	55.0	65.3

Table 3.13: English evaluation results on publicly available non-news corpora.

While HeidelTime has been the first temporal tagger performing domain-sensitive temporal tagging, the first non-news-style corpus annotated with temporal expressions was developed by Mazur and Dale (2010). The WikiWars corpus contains parts of Wikipedia articles about important wars in history (cf. Section 3.2.3). The developers evaluated their TIMEX2-compliant temporal tagger DANTE on WikiWars using an initial and an improved rule set for which they used the ACE TERN 2005 training data as well as the WikiWars corpus. HeidelTime's and DANTE's evaluation results on WikiWars are shown in Table 3.13(a). Although Mazur and Dale (2010) used the WikiWars documents for improving DANTE, they were not able to achieve promising normalization results on the Wikipedia articles since "the strategy of using the document time stamp for the interpretation of context-dependent expressions does not work at all for WikiWars documents" (Mazur and Dale, 2010). In contrast, using HeidelTime's narrative-style normalization strategy works very well so that HeidelTime significantly outperforms DANTE's initial and improved rule sets although HeidelTime annotates temporal expressions following TimeML and WikiWars was not used during HeidelTime's development process.[47]

In addition to addressing the challenges of temporal tagging news- and narrative-style documents, we also performed experiments and developed initial resources and normalization strategies for processing colloquial and autonomic documents (cf. Section 3.3). HeidelTime's evaluation results on these two domains are presented in Table 3.13(b) and Table 3.13(c). Despite the challenges of these two domains, the results look promising. Unfortunately, we cannot compare our evaluation results on these domains to the results of other temporal taggers since HeidelTime is the only temporal tagger explicitly addressing these domains. However, in the following section, we show the value of domain-sensitive temporal tagging by presenting the results of our cross-domain evaluation experiment.

[47] In the meanwhile, DANTE follows HeidelTime's approach to use different normalization strategies depending on the domain of the documents to be processed (for details, see Mazur, 2012).

corpus (domain)	strategy	relaxed P	relaxed R	relaxed F1	strict P	strict R	strict F1	value P	value R	value F1	relaxed+value P	relaxed+value R	relaxed+value F1	strict+value P	strict+value R	strict+value F1
TimeBank (news)	**news**	90.7	91.5	**91.1**	83.7	84.4	**84.1**	86.2	86.2	**86.2**	78.3	78.9	**78.6**	73.5	74.1	**73.8**
	narrat.	90.7	91.5	**91.1**	83.7	84.4	**84.1**	67.5	67.5	67.5	61.2	61.7	61.5	57.5	58.0	57.7
	colloq.	90.5	91.7	**91.1**	82.8	83.9	83.4	86.0	86.0	86.0	77.9	78.9	78.4	72.4	73.4	72.9
	scient.	90.7	91.5	**91.1**	83.0	83.7	83.4	81.2	81.2	81.2	73.7	74.3	74.0	69.0	69.6	69.3
WikiWars (narrative)	news	93.9	82.6	87.9	86.0	75.7	**80.5**	64.7	65.1	64.9	60.7	53.4	56.9	57.6	50.7	53.9
	narrat.	93.9	82.6	87.9	86.0	75.7	**80.5**	89.5	90.1	**89.8**	84.1	73.9	**78.7**	79.6	70.0	**74.5**
	colloq.	93.3	83.4	**88.1**	84.3	75.3	79.6	64.1	64.5	64.3	59.8	53.5	56.5	56.0	50.0	52.8
	scient.	93.9	82.9	88.0	85.5	75.4	80.1	63.8	64.2	64.0	59.9	52.9	56.2	56.7	50.1	53.2
Time4SMS (colloquial)	news	99.3	85.2	91.7	98.9	84.8	91.3	97.9	97.9	**97.9**	97.2	83.4	89.8	97.2	83.3	89.7
	narrat.	99.3	85.2	91.7	98.9	84.8	91.3	96.4	96.4	96.4	95.7	82.1	88.4	95.6	82.0	88.3
	colloq.	99.4	91.1	**95.1**	98.1	90.0	**93.9**	97.1	97.1	97.1	96.4	88.5	**92.3**	96.0	88.1	**91.9**
	scient.	99.3	85.3	91.8	98.8	84.8	91.3	97.8	97.8	97.8	97.2	83.4	89.8	97.1	83.3	89.7
Time4SCI (scientific)	news	95.1	55.0	69.7	76.2	44.1	55.8	74.4	74.4	74.4	70.8	40.9	51.9	67.6	39.1	49.5
	narrat.	95.1	55.0	69.7	76.2	44.1	55.8	74.4	74.4	74.4	70.8	40.9	51.9	67.6	39.1	49.5
	colloq.	95.0	59.1	72.8	75.9	47.2	58.2	75.7	75.7	75.7	71.9	44.7	55.1	67.8	42.2	52.0
	scient.	95.1	66.6	**78.3**	87.9	61.6	**72.4**	88.7	88.7	**88.7**	84.4	59.1	**69.5**	78.6	55.0	**64.7**

Table 3.14: Evaluating HeidelTime using different domain settings on corpora of the four domains.

3.6.5 Cross-domain Evaluation

For our cross-domain evaluation (Strötgen and Gertz, 2012b), we used four corpora of the four domains that HeidelTime distinguishes: Timebank (news), WikiWars (narrative), Time4SMS (colloquial), and Time4SCI (scientific). On each corpus, we ran HeidelTime with the four different domain settings. The results of this cross-domain evaluation experiment are presented in Table 3.14.

Since HeidelTime's news and narrative domain settings only differ with respect to the normalization strategy for relative and underspecified expressions, the extraction results for these two strategies are identical on all four corpora. However, the most important finding of the cross-domain evaluation experiment is that using HeidelTime's narrative normalization strategy outperforms HeidelTime's news normalization strategy on WikiWars by more than 20 percentage points in f_1-score (relaxed+value). These 20 percentage points performance would be lost using a news-style temporal tagger on narrative-style documents such as Wikipedia articles. Note that all other temporal taggers that are publicly available are developed for processing news-style documents. This should be kept in mind when a temporal tagger is used to process narrative-style documents.[48]

In a similar way, as the narrative strategy outperforms the news strategy on the narrative corpus, the news strategy outperforms the narrative strategy on the news corpus. In addition to the fact that the correct domain settings always outperform the other domain settings on the corresponding corpora, the results in Table 3.14 demonstrate that it is useful to extend the patterns, normalization information, and rules for the colloquial and also for the scientific (autonomic) domain. Not only the normalization results (value, relaxed+value, strict+value) on Time4SMS and Time4SCI are improved using the colloquial and scientific (autonomic) domain settings, respectively, but also the extraction results (relaxed and strict).

[48] As mentioned above, the DANTE's latest version now also uses different normalization strategies for news and narrative.

In summary, the cross-domain evaluation experiment illustrates the value of HeidelTime's domain-sensitive temporal tagging approach. Due to the lack of publicly available corpora for other languages covering more than one domain, we performed this cross-domain evaluation for English only. However, we also evaluated HeidelTime on corpora of other languages as will be described in the next section.

3.6.6 Further Results on Non-English Corpora

In this section, we present HeidelTime's evaluation results on non-English corpora. The results are summarized in Table 3.15.

Spanish Evaluation Results

In Table 3.15(a), HeidelTime's evaluation results for Spanish are presented. There is only one temporally annotated corpus for Spanish, namely the Spanish TimeBank corpus (Saurí and Badia, 2012), which was split in the context of the TempEval-3 challenge into a training set and an evaluation set (UzZaman et al., 2013). While we compared HeidelTime's performance on the evaluation set to other temporal taggers in Section 3.6.3, we here present the results on the training corpus additionally and show all evaluation measures. Note that the training corpus was used to develop the Spanish resources (cf. Section 3.5.7). On both sets, HeidelTime achieves high quality results for the extraction and the normalization tasks.

Italian Evaluation Results

The Italian evaluation results on the TempEval-2 data are presented in Table 3.15(b). Since there are no evaluation results of other temporal taggers performing the original, token-level TempEval-2 evaluation, we transformed the documents into the TempEval-3 format and used the TempEval-3 evaluation script to calculate the evaluation measures. The expression-level evaluation is more frequently used, more intuitive, and, in addition, allows a better comparison between the evaluation results of the different languages.

Compared to the Spanish evaluation results, HeidelTime achieves lower results on both, the training and the test data sets. In addition, although we developed the Italian HeidelTime resources using the TempEval-2 training corpus (Strötgen et al., 2014a), HeidelTime achieves better results on the test set than on the training set. While we present a more detailed error analysis in Section 3.6.8, one of the reasons is that there are several annotation errors in the gold standard, in particular in the Italian training set, and thus the lower precision on the Italian documents can be explained by several missing annotations.

Table 3.15(c) shows HeidelTime's evaluation results on the I-CAB training and evaluation sets. In addition, we show the results of the participants of the EVALITA challenge on the test set (Bartalesi Lenzi and Sprugnoli, 2007). Note that the corpus contains TIMEX2 annotations, and that we did not use the training corpus for developing HeidelTime's Italian resources. Nevertheless, our results are competitive, although the best system of the EVALITA challenge (Negri, 2007) achieves much better results.

HeidelTime's results on the TempEval-2 data are much better than on the I-CAB corpus. In addition to the differences between TIMEX2 and TIMEX3 annotations, a first error analysis showed that our resources fail to match many time expressions and durations of small granularity, which were not frequent in the TempEval-2 data but very frequent in the I-CAB corpus. Further details of the error analysis will be presented in Section 3.6.8.

(a) Spanish evaluation results: TempEval-3 training and test data.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
training	96.2	86.4	91.0	90.5	81.3	85.7	92.4	92.4	92.4	88.9	79.8	84.1	83.6	75.1	79.1
test	96.0	84.9	90.1	90.9	80.4	85.3	94.7	94.7	94.7	90.9	80.4	85.3	86.1	76.1	80.8

(b) Italian evaluation results: TempEval-2 training and test data (expression-based evaluation).

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
training	90.0	82.6	86.1	82.5	75.7	79.0	89.4	89.4	89.4	80.4	73.8	77.0	73.7	67.7	70.6
test	92.2	84.9	88.4	87.1	80.2	83.5	93.5	93.5	93.5	86.2	79.4	82.6	81.4	74.9	78.0

(c) Italian evaluation results: I-CAB training and test data.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
training	95.5	62.9	75.9	68.5	45.1	54.4	70.1	71.2	70.7	67.0	44.1	53.2	48.2	31.7	38.3
test	94.6	61.3	74.4	65.0	42.1	51.1	77.7	78.2	77.9	73.5	47.6	57.8	52.8	34.2	41.5
FBKirst_Negri	95.7	89.8	92.6							68.5	63.3	67.4			
UniPg_Faina	77.7	70.3	73.8							24.9	19.6	21.9			
UniAli_Puchol	78.4	67.4	72.5												
UniAli_Saquete	82.5	53.2	64.7							51.5	35.6	42.1			

Results of the other systems according to Bartalesi Lenzi and Sprugnoli (2007).

(d) Arabic evaluation results: Arabic train-203, test-150, test-50, and test-50*.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
train-203	80.6	91.0	85.5	63.4	71.6	67.2									
test-150	81.0	89.6	85.1	65.4	72.3	68.7									
test-50	81.3	90.0	85.5	63.3	70.1	66.5									
test-50*	93.2	90.5	91.8	84.3	81.8	83.0	90.0	90.0	90.0	83.8	81.4	82.6	75.8	73.6	74.7

(e) French evaluation results: French TimeBank.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	92.0	89.4	90.7	86.4	84.0	85.2	80.0	80.0	80.0	73.6	71.5	72.6	69.2	67.2	68.2

(f) German evaluation results: WikiWarsDE.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	98.4	84.5	90.9	92.6	79.5	85.6	87.3	87.3	87.3	86.0	73.8	79.4	82.7	71.0	76.4

(g) Vietnamese evaluation results: WikiWarsVN.

	relaxed			strict			value			relaxed+value			strict+value		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
HeidelTime 1.5	100.0	98.2	99.1	94.4	92.7	93.6	92.1	92.1	92.1	92.1	90.5	91.3	87.0	85.4	86.2

Table 3.15: Evaluation results on publicly available non-English corpora (HeidelTime version 1.5).

Arabic Evaluation Results

Table 3.15(d) shows HeidelTime's evaluation results on the Arabic documents of the ACE 2005 training corpus. The original corpus contains TIMEX2 annotations, however, only the extents of temporal expressions are annotated, i.e., no normalization information is provided (cf. Section 3.4.3). As described in Section 3.4.5, we split the corpus into a training and two test sets: training-203, test-150, and test-50, and re-annotated the documents of the test-50 set using TIMEX3 annotations (test-50*). During the re-annotation process, we also manually added normalization information (value attribute) to the annotations. Since there were many inconsistencies in the annotation of the original extents and many missing temporal expressions, we also corrected existing and added missing annotations.

While the extraction performance on the training set and the two test sets are already promising, the results on the test-50* corpus are even better since the documents now contain TIMEX3 annotations and annotation errors in the gold standard are removed. More importantly, although we cannot compare HeidelTime's evaluation results due to the lack of other temporal taggers for Arabic that extract and normalize temporal expressions, the evaluation results look very promising. HeidelTime does not only achieve high quality extraction results for Arabic (91.8 f_1-score for relaxed matching) but also high quality normalization results (82.6 f_1-score for relaxed matching with correct value normalization). While Saleh et al. (2011) also report extraction results on the ACE 2005 training corpus, their results are not comparable to our results since their tool extracts temporal phrases not very similar to TIMEX2 or TIMEX3 annotations. Their reported f_1-scores on the whole corpus are 43.3 for relaxed and 24.0 for strict matching.

German Evaluation Results

Since there was no temporally annotated corpus for German, we had to develop an own corpus to evaluate HeidelTime's quality for processing German documents. As described in Section 3.4.5, we developed the WikiWarsDE corpus (Strötgen and Gertz, 2011) and Table 3.15(f) contains HeidelTime's evaluation results on this corpus. Note that the corpus contains Wikipedia articles, i.e., HeidelTime is used with its narrative domain settings. Thus, the results should not be directly compared to the results for the other languages. Nevertheless, the results look promising. Unfortunately, there are no other temporal taggers for German available that could have been used for comparison.

Vietnamese Evaluation Results

Similar as for German, there was no Vietnamese temporally annotated corpus available so far. Thus, we developed WikiWarsVN and used it to evaluate HeidelTime's temporal tagging quality for Vietnamese. In Table 3.15(g), the evaluation results are presented. The Vietnamese evaluation results are much better than HeidelTime's German results, probably mainly due to the much simpler language characteristics of Vietnamese. In addition, the Vietnamese WikiWarsVN documents are much shorter and the temporal expressions are less challenging to normalize than in the English and German articles.

French Evaluation Results

The French HeidelTime resources were developed by Moriceau and Tannier from LIMSI, Paris (Moriceau and Tannier, 2014) and evaluated using the French TimeBank corpus (Bittar et al., 2011). The evaluation results are shown in Table 3.15(e) and look very promising. This demonstrates that one does not have to be involved in HeidelTime's development to create high quality HeidelTime resources.

Summary

In summary, the non-English evaluation results demonstrate HeidelTime's high quality for temporal tagging documents of several languages. Thus, we can conclude that the development of HeidelTime as a multilingual, cross-domain temporal tagger opens up opportunities for natural language processing research relying on temporal information in multiple languages.

3.6.7 Processing Time Performance

While fast processing performance has not been the major goal during the development of HeidelTime, we still payed attention to this issue to allow the usage of HeidelTime in large-scale document processing scenarios.

In this section, we report on HeidelTime's processing time performance by presenting time measurements of processing two types of corpora. (i) To present HeidelTime's processing time performance with different language and domain settings, we process the gold standard corpora used for evaluating HeidelTime. (ii) We report time measurements for processing Wikipedia in different languages as representatives of large document collections. In both settings, HeidelTime's UIMA version is used. For each run, we will thus report the processing times of the whole workflow and of the HeidelTime analysis engine separately. In both cases, the time measurements are directly reported by the UIMA framework.

Processing Time Performance on Evaluation Corpora

Since the evaluation corpora have different formats and languages, we process them using HeidelTime's UIMA version in combination with the collection readers, analysis engines, and CAS consumers being part of the UIMA HeidelTime kit. The workflows for all corpora are depicted in Figure 3.9. For instance, the English news-style corpora in TempEval-3 format are read by the TempEval-3 reader and processed by the TreeTagger wrapper analysis engine for linguistic preprocessing (sentence splitting, tokenization, and part-of-speech tagging). Then, HeidelTime is applied with its news-style strategy, and the TempEval-3 writer outputs the results in the format required by the TempEval-3 evaluation scripts.

Due to the relatively small sizes of all evaluation corpora, we ran the workflows on a standard laptop (Intel dual core P8700 2.53 GHz, 4 GB ram) without parallelization or any other kind of tuning.

In Table 3.16, the time performance measurements are shown for all evaluation corpora – ordered by language and number of tokens (token count according to the number of tokens extracted by our UIMA wrappers for linguistic preprocessing). The processing times of the whole workflows range from less than six seconds to almost 520 seconds for the smallest and largest corpora with respect to the number of tokens (TE-2 English and ACE TERN 2004), respectively. Note that the workflow processing time is not only sensitive to the total number of tokens but also to the number of documents. While processing the Time4SMS corpus that contains many short documents is rather slow, the WikiWars and WikiWarsDE corpora contain only few but quite long documents and are processed much faster.

The processing time of the HeidelTime analysis engine is rather less sensitive to the number of documents but mostly depends on the total size of the corpora (total number of tokens). In Figure 3.10, the processing times of the UIMA workflows and the HeidelTime analysis engines are depicted. In both figures – note the log scale of the x-axes – the dotted line represents a linear regression based on the

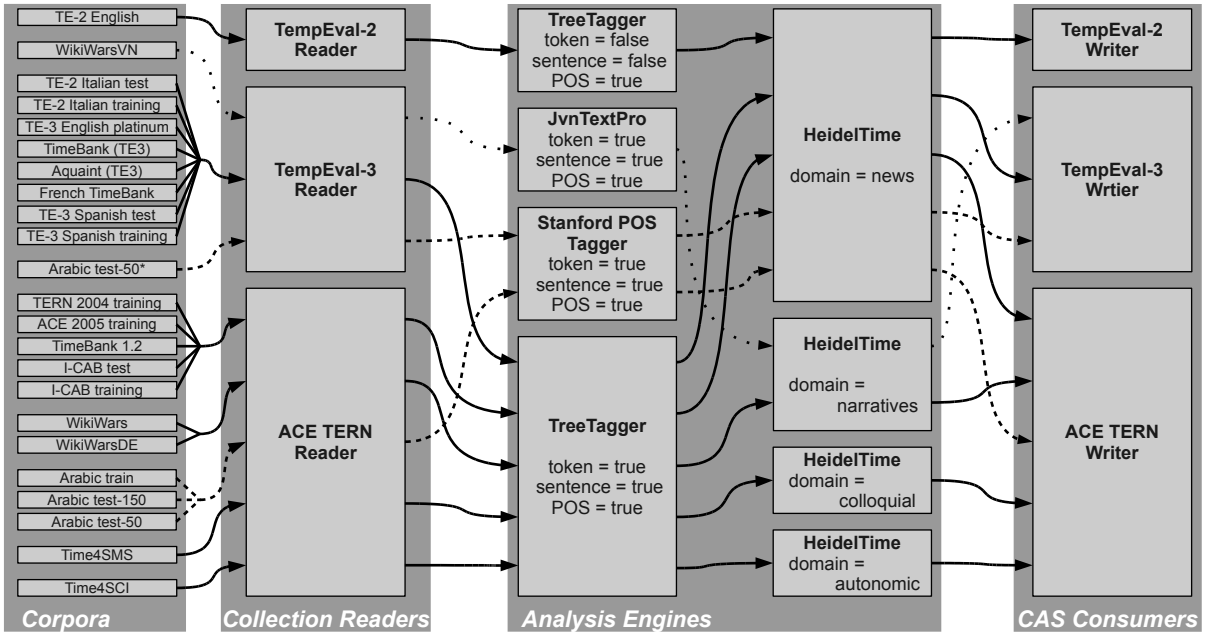


Figure 3.9: UIMA workflows to measure time performance on evaluation corpora.

processing times for all corpora. For the HeidelTime analysis engine, the linear regression is a quite good estimation with an asymptotic standard error of just 3.3%. In contrast, the linear regression is a less well estimation for the full UIMA workflows (asymptotic standard error 16.5%).

In addition to the document length sensitivity – represented by the WikiWars and Time4SMS corpora explicitly marked in Figure 3.10(a) – the language also plays a more significant role for the processing time of the whole workflows than for the processing time of the HeidelTime analysis engines. For example, processing Italian is slower than processing English. These differences are mainly due to performance differences for linguistic preprocessing performed by our UIMA wrappers. In general, these could be improved to reduce the processing times of the whole workflows since they currently require a lot of I/O for each preprocessing subtask (sentence splitting, tokenization, and part-of-speech tagging). Avoiding I/O would make the preprocessing faster since the preprocessing tasks themselves are quite fast. However, due to the rather minor role of performance for our work, this will be addressed in future work.

Processing Time Performance on Wikipedia

After having presented processing times of the rather small evaluation corpora, we now report time performance measurements for processing the English and Spanish Wikipedia to demonstrate that HeidelTime can easily be used to process large document collections.

Except of language settings, the workflows of both document collections are identical. Since we stored the text parts of all Wikipedia articles in a NoSQL MongoDB database, we used a simple MongoDB collection reader. For linguistic preprocessing, our UIMA TreeTagger wrapper is used with the respective language models. Then, HeidelTime is applied with its narrative normalization strategy before a CAS consumer counts sentences, tokens, and temporal expressions extracted from the documents.

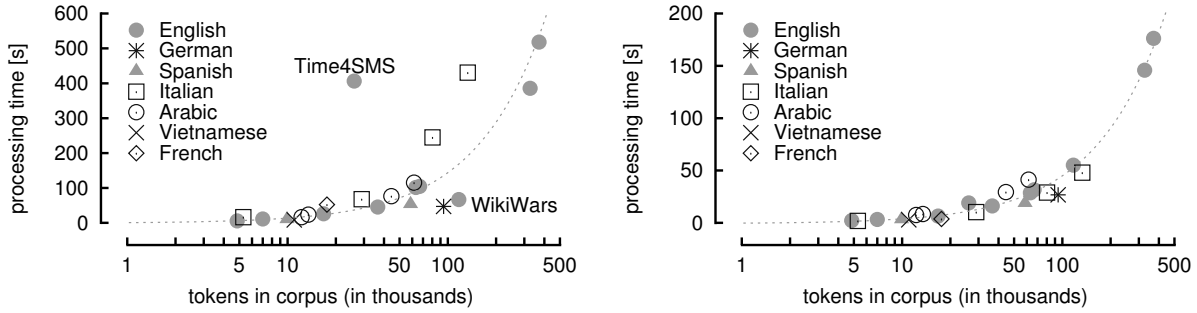
corpus	language	domain	docs	tokens	workflow [s]	HeidelTime [s]
TE-2 test	English	news	9	4,849	5.7	2.2
TE-3 platinum	English	news	20	7,000	11.2	3.2
Time4SCI	English	autonomic	50	16,760	26.2	6.5
Time4SMS	English	colloquial	1,000	26,054	406.3	19.0
Aquaint TE3	English	news	73	36,497	45.4	16.2
TimeBank TE3	English	news	183	63,173	101.3	28.7
TimeBank 1.2	English	news	183	66,628	104.4	31.6
WikiWars	English	narratives	22	117,169	66.9	55.1
ACE 2005	English	news	599	325,974	385.8	145.9
ACE TERN 2004	English	news	862	370,964	518.0	176.3
TE-2 test	Italian	news	13	5,293	16.4	1.9
TE-2 training	Italian	news	51	28,988	67.9	10.3
I-CAB test	Italian	news	190	80,293	244.9	29.0
I-CAB train	Italian	news	335	133,032	430.5	48.0
ACE train-50*	Arabic	news	50	12,228	16.3	7.5
ACE train-50	Arabic	news	50	13,489	23.7	8.4
ACE test-150	Arabic	news	150	44,449	76.5	29.4
ACE train-203	Arabic	news	203	61,494	115.4	41.3
TE-3 test	Spanish	news	35	9,914	10.3	3.2
TE-3 training	Spanish	news	150	58,493	53.2	18.8
WikiWarsDE	German	narratives	22	94,058	47.3	26.8
WikiWarsVN	Vietnamese	narratives	15	11,014	8.3	2.9
TimeBank-FR	French	news	108	17,611	52.4	3.7

Table 3.16: HeidelTime's processing time performance on evaluation corpora.

In Table 3.17, we report some information about the Wikipedia dumps in addition to the time performance measures for the full UIMA workflows and the HeidelTime analysis engines. Note that in contrast to the experiments on the evaluation corpora, we used an Intel quad-core i7-4770 (3.40GHz, 16 GB ram) and multi-threading by setting the CAS pool size and processing unit thread count parameters to 16 each.

Processing the almost 4.5 million documents of the English Wikipedia with more than 1,708 million tokens took in total 71 hours. About 23% of the time was used by the HeidelTime analysis engine itself. Processing the Spanish workflow was faster with respect to both, per document and per 1000 tokens measurements. The processing time for the HeidelTime analysis engine was also slightly faster for processing the Spanish Wikipedia than for processing the English Wikipedia, but the main difference between processing the English and Spanish Wikipedia is due to faster linguistic preprocessing the Spanish documents. This is also the reason why the HeidelTime analysis engine took about 33% of the full processing time of the Spanish Wikipedia.

Given the facts that only a single machine was used in these performance measurements and that there is no need for any manual effort to run HeidelTime in parallel mode, these numbers demonstrate that HeidelTime can be used out-of-the-box to process large-scale document collections.



(a) Full workflows. (b) HeidelTime only.

Figure 3.10: HeidelTime's processing time performance on all evaluation corpora.

language	documents	tokens	total time [h] workflow / HeidelTime	per document [s] workflow / HeidelTime	per 1000 tokens [s] workflow / HeidelTime
English	4,457,716	1,708,494,667	70.6 / 16.4 (23%)	0.057 / 0.013	0.149 / 0.0345
Spanish	1,035,680	409,180,706	9.0 / 3.02 (33%)	0.031 / 0.011	0.079 / 0.0266

Table 3.17: HeidelTime's processing time performance on English and Spanish Wikipedia.

3.6.8 Error Analysis

In order to be able to better interpret HeidelTime's evaluation results presented in the previous sections, we performed a detailed error analysis (see also Strötgen et al., 2013, 2014a). The most important findings and conclusions are summarized in the following.

Error Types

In general, four types of errors can be distinguished:

- false negatives (FNs): annotated in the corpus but not extracted by the system.
- false positives (FPs): extracted by the system but not annotated in the corpus.
- relaxed matching only (RMs): either only parts of an expression in the corpus are extracted or only parts of an extracted expression are annotated in the corpus.
- incorrect value normalization (IVs): expressions that are (partially) extracted but the normalized value information of the system and the corpus is different.

Intended False Negatives (FNs)

Except for Vietnamese, for a rule-based approach, HeidelTime's recall is relatively low. Thus, there are several false negatives. However, due to our focus on the correct normalization of temporal expressions, i.e., high *relaxed + value* scores, some false negatives are intentional. We do not want to extract temporal expressions that are unlikely to be normalized correctly. Expressions such as *some time* or *the latest period*, for example, can refer to seconds or years depending on the context. With correct normalization

being the main goal, we prefer to not extract such expressions that cannot be normalized correctly with high probability. For example, in the English[49] and Spanish TempEval-3 training corpora, 55% and 29% of 117 and 149 false negatives are due to such imprecise expressions.

Trade-off between Precision and Recall (FNs and FPs)

In all languages, there is a trade-off between precision and recall due to expressions referring to past, present, or future (normalized as PAST_REF, PRESENT_REF, and FUTURE_REF, respectively). These expressions, e.g., *now* and *recent*, are either only annotated in some contexts or inconsistently in the gold standards. Thus, depending on whether or not such expressions are included in HeidelTime's resources, they are responsible for false positives or false negatives, respectively. Note that for several tasks such as temporal information retrieval, i.e., querying with temporal constraints, such temporal expressions can be considered as less important anyway since they are not normalized to any specific point in time.

Differences between TIMEX2 and TIMEX3 (RMs and IVs)

Some of HeidelTime's errors occurring in the TIMEX2-annotated corpora can be explained by the differences between TIMEX2 and TIMEX3. For instance, there are several event-anchored temporal expressions in the WikiWars corpus, in the Arabic part of the ACE 2005 multilingual training corpus, and in the Italian I-CAB corpus. As explained in Section 3.2.1, events can be part of TIMEX2-annotated temporal expressions while they are usually outside of temporal expressions according to TimeML. Since HeidelTime is annotating temporal expressions according to TIMEX3, we do not try to extract event-anchored temporal expressions. Thus, although TIMEX2 and TIMEX3 are similar, such expressions diminish HeidelTime's evaluation results on the TIMEX2-annotated corpora. More precisely, these differences between TIMEX2 and TIMEX3 resulted in both, relaxed matching only errors and in incorrect value normalization errors.

This error source also partially causes the big differences between HeidelTime's evaluation results on the Italian TempEval-2 and the I-CAB corpora. Furthermore, these errors explain the large differences between strict and relaxed precision on the Italian I-CAB, the WikiWars, and the Arabic TIMEX2-annotated corpora.

Annotation Errors in the Corpora (all error types)

A forth type of error that cannot be addressed by simply improving the temporal tagger are all kinds of annotation errors in the gold standard corpora such as missing annotations or incorrect value information.

For example, in the Italian TempEval-2 training data, missing annotations are quite frequent, resulting in a much lower precision of our Italian resources on the training corpus than on the evaluation corpus. The latter seems to be annotated more thoroughly. Examples of incorrect normalization information in gold standard annotations are imprecise annotations. Expressions such as *last week* and *next week* are often not annotated to the week the expressions actually refer to but to an unspecific week using "WXX". Many examples of such annotations can be found in the TimeBank corpus, for instance.

[49] After the TempEval-3 competition, we performed an error analysis on the English and Spanish training and test sets. Note, however, that we only used the TimeBank corpus as the representative of the English training set (Strötgen et al., 2013).

Sometimes it is possible to normalize temporal expressions in different ways. For example, *twelve months* can be annotated as "P12M" (a twelve months duration) or as "P1Y" (a one year duration). Although such value pairs carry the same meaning, they are evaluated as incorrect value normalizations by the annotation scripts. Note that such issues are not necessarily annotation errors but they occur due to inconsistencies in the manual annotations. In addition, there are also some further inconsistencies in most of the gold standard corpora with respect to the extents of temporal expressions. For example, determiners and modifiers are sometimes annotated as parts of temporal expressions and sometimes not.

Incorrectly Detected Reference Times (IVs)

A further main source for incorrect value normalization of underspecified expressions (*Feb. 1*; *Monday*) are incorrectly detected reference times. In Section 3.3.8, we presented HeidelTime's strategies to identify the correct reference time but these are not always correct. For instance, in narrative-style documents, the reference time of an underspecified expression is the previously mentioned temporal expression of a fitting granularity. However, this decision may be incorrect as in the example shown in Figure 3.2(b) (page 49). Here, *December 27* would be normalized to "1978-12-27" instead of "1979-12-27" due to the previously mentioned expression *1978* in the document's text. While it might be possible to exclude "1978" from the set of possible reference times due to its attributive usage, such complex and further complex reference time detection methods would have to be carefully evaluated and are not yet included in HeidelTime. Note that in the example document, the next expression *the morning* would then also be normalized incorrectly although the determined reference time (December 27) is correct. Thus, a single wrong decision can result in multiple normalization errors.

Similarly, the method for determining the reference time of underspecified expressions in news-style documents may be misleading, too. As was shown in the comparative corpus analysis (cf. Section 3.3.7), sometimes the reference time of underspecified expressions has to be identified in the documents' texts even in news-style documents, although HeidelTime's strategy of using the document creation time as reference time is correct in most cases.

Incorrectly Detected Relations to Reference Times (IVs)

A similar challenge and source of incorrect value normalizations are incorrectly determined relations to reference times (e.g., due to false tense identification). Except for Vietnamese, incorrect tense identification results in normalization errors for underspecified expressions in news- and colloquial-style documents. Thus, a more sophisticated tense identification than just relying on part-of-speech tags and some regular expressions – as in the case of Spanish tense identification (Strötgen et al., 2013) – would be helpful.

However, tense identification does not always help to correctly identify the relation to the reference time as demonstrated in the following example from the Arabic corpus: ستصدر محضرا عن جلسة الاثنين (It will issue a report about Monday meeting). While future tense was identified in the sentence, "Monday" refers to the current Monday. However, to correctly identify the relation to the reference time in such cases, a deeper linguistic analysis would be necessary. In addition, without any further context information, it cannot be decided whether "Monday" refers to a future, the current, or a past Monday. Although this example is selected from the Arabic corpus, such problems occur in many languages.

Complex and Missing Patterns (all error types)

In general, complex temporal expressions can result in a combination of several errors in the evaluation. An example from the Italian corpus where a relaxed matching results in an incorrect value normalization error and a false positive additionally is the expression *alle ore 11 del mattino di oggi* (today morning at 11 am), which should be annotated as "2004-09-07T11:00". Instead of the whole expression, HeidelTime's Italian resources result in a relaxed match "mattino di oggi" (today morning) with a false value "2004-09-07-TMO" and a false positive "ore 11" (11 am). Note that the latter expression has the correct value, which, however, is not considered by the evaluation scripts due to the first overlapping expression.

More general rules, e.g., allowing several tokens of specific part-of-speech tags between other tokens in the extraction part of a rule, could help to extract more complex temporal expressions. However, such rules then may result in incorrectly extracted and incorrectly normalized temporal expressions. Thus, when developing rules, it is often a trade-off and the goal is to write the extraction parts of the rules as generally as possible and as precisely as necessary.

In addition to unmatched complex patterns, there are also some errors due to relatively simple missing patterns in the HeidelTime resources. For example, there is some room for improving HeidelTime's resources to extract time and duration expressions – in particular durations of small granularities. While resources for such expressions could probably be improved for HeidelTime resources for all supported languages, in our error analysis, we mainly identified such errors in the Italian I-CAB corpus. This corpus contains many news documents from the sports genre in which expressions referring to a specific minute in a match (e.g., *al 4'*) or to a duration of an event (e.g., *13'35"64*) are frequent. Since articles from the sports genre are not well covered in neither of the corpora used for developing HeidelTime resources, we have not added rules for such expressions so far. In addition to several miscellaneous expressions currently not extracted, there is also a specific group of temporal expressions that HeidelTime does not deal with so far, as described next.

Temporal Expressions Referring to Historic Dates (FNs, PMs, and IVs)

HeidelTime's current version (version 1.5) does not support the normalization of temporal expressions referring to historic dates (dates before the year 1000). While in news-style documents such expressions rarely occur, in narrative-style documents such expressions are quite frequent and occur in documents about history. For example, in the WikiWars corpus, two of the 22 documents are about wars that took place between 500 and 150 Before Christ, and thus contain many temporal expressions referring to dates in this time interval. Depending on specific temporal expressions, the lack of dealing with historic date expressions results in false negatives (e.g., *300 BC* would not be extracted) or partial matches and incorrect value normalizations (*2000* in "2000 BC"). Furthermore, relative temporal expressions (e.g., *one year later*) would be normalized incorrectly since there is no chance that an expression referring to a year before 1000 is selected as reference time.

Note that in contrast to other complex temporal expressions, addressing the extraction and normalization of temporal expressions referring to historic dates cannot by solved by simple adding some more rules to HeidelTime's resources. In contrast, the normalization of relative expressions would have to be modified in the source code, e.g., for calculating the values of expressions such as *the following year*, and in general, to handle the prefix "BC" in the value attribute. More details on this issue are described in Section 3.7, where possible extensions for HeidelTime are discussed.

Expressions Referring to Non-standard Calendars (FNs)

In the Arabic corpus, there is an additional error source for false negatives, namely temporal expressions referring to the Hijri calendar. Such expressions are not included in the Arabic HeidelTime resources yet. While their extraction would not be very difficult, their normalization is challenging and not solvable without adding further normalization functions to HeidelTime's source code.

In general, HeidelTime currently does not perform any normalizations to other calendars than the Gregorian calendar and temporal expressions based on other calendars are not normalized correctly.

Ambiguities (FPs, FNs, and IVs)

In some languages, there are words that can carry a temporal meaning and, in addition, a non-temporal meaning. For instance, in English, "March" and "May" are month names but can also be used as verbs, auxiliary verbs, or even as nouns without a temporal meaning (e.g., *March* in "March of the Iron Will" in one of the WikiWars documents). In Arabic, قرن (/qrn/) can mean either "century" or "horn" and الاثنين (/al'ithnayn/) either "Monday" or "two". While ambiguous words are rather rare in the corpora, numbers sometimes also do not refer to temporal expressions although they are mentioned in similar contexts, e.g., in news documents from the sports genre, results of sporting events such as "7-6".

Another type of ambiguities mainly occurred in the Italian corpora. Here, two-digit numbers referring to year expressions resulted in several false negatives. For such expressions (e.g., *99*), the context surrounding the two-digit numbers would have to be analyzed carefully to avoid the extraction of numbers not referring to year expressions. In addition, if narrative-style documents were to be processed, it would be necessary to determine whether such two-digit numbers refer to the specific year AD, BC, or any other century as the current or previous centuries. Such rules are not added to the Italian resources yet. However, note that temporal expressions consisting only of a two-digit number occurred almost only in the Italian corpora.

Another type of ambiguity results in difficulties for a correct value normalization since some expressions may refer to different types of temporal expressions. For instance, the expression *the year* can either refer to a specific year (i.e., as a date expression) or to a duration with the length of one year. Similarly, temporal expressions that can refer to different granularities are challenging. Such granularity-related errors are discussed next.

Granularity Errors (IVs)

For some relative temporal expressions, it is difficult to determine to a date of which granularity they refer. For instance, the expression *a year ago* can either refer to a day, month, quarter, or year. To avoid evaluation errors due to such granularity issues, the manual annotation and the system's annotation have to be identical – although sometimes it might be even difficult to decide for a reader with full context understanding which granularity fits best.

Errors in Colloquial- and Autonomic-style Documents (all error types)

In addition to the errors described so far, there are some errors related to the specific challenges occurring in colloquial- and autonomic-style documents. However, since we used our newly developed corpora for adapting HeidelTime to these domains, we cannot provide a detailed error analysis for these domains.

Figure 3.11: A map showing from which countries the HeidelTime Google Code project site (`http://code.google.com/p/heideltime/`) was accessed; according to Google Analytics for the time period between January 1, 2013 and July 8, 2014.

Summary

While some of the errors described in the error analysis can be addressed with relatively low effort, e.g., by adding specific patterns and rules, other error types are more difficult to address. Furthermore, there are also a couple of issues diminishing HeidelTime's evaluation results, which are independent of HeidelTime-related errors. In the next section, we will discuss which of the error sources will be addressed in the future and, in general, how HeidelTime can be further developed.

3.7 HeidelTime in the Future

By making HeidelTime publicly available, we already made important contributions to the research community. Exemplarily, Figure 3.11 shows from which countries our HeidelTime project website was accessed. In addition, as mentioned at the beginning of Section 3.5, HeidelTime is already used by several research groups.[50] However, we do not consider HeidelTime as a final system for which no further improvements can be reached. In the following, we describe our plans to further increase HeidelTime's value for the research community.

Improving Resources of Currently Supported Languages

As described above, we used several existing temporally annotated corpora for evaluation purposes only and not for developing or improving HeidelTime resources. However, some of HeidelTime's temporal tagging errors on these corpora are due to missing patterns or rules as was shown in the error analysis. For HeidelTime's future versions, we plan to exploit the results of the error analysis and improve HeidelTime's language resources by adding missing patterns and rules based on the currently not correctly extracted expressions.

[50] The citation counts of our HeidelTime-related research papers further demonstrate their impact. For instance, Strötgen and Gertz (2010a) and Strötgen and Gertz (2013a) are cited 94 and 47 times, respectively, according to Google Scholar (December 10, 2014).

While the corpora we used for evaluation purposes so far then cannot be used anymore to compare HeidelTime's quality to the quality of other taggers, which did not use the corpora as training or development data, this will further improve HeidelTime's extraction and normalization quality.

Temporal Expressions Referring to Historic Dates

As described in the error analysis (Section 3.6.8) HeidelTime's current version does not detect and normalize temporal expressions referring to dates before the year 1000 or even to years Before Christ (BC). While one could easily add a couple of rules to detect some date expressions referring to historic dates, there are some challenges that have to be kept in mind if one wants to complete the task more accurately. For instance, depending on the context, expressions such as *in the year 90* can either refer to 90 AD or 90 BC (in documents about history) or to 1990 (e.g., in current newspaper articles).

The detection of temporal expressions referring to historic dates is important when processing narrative-style documents which are often documents about history. Thus, despite the difficulties, we are currently addressing this issue, and HeidelTime will also be able to detect and normalize such expressions.[51]

Support for Further Languages

In addition to our hope that other researchers develop further language resources for HeidelTime as already done for Dutch and French, we also plan to develop HeidelTime language resources for additional languages. We are particularly interested in wide-spread languages for which a lot of written documents are available. Examples are Chinese, Hindi, and Russian.[52] However, to bring forward research in the context of the humanities, we are also interested in performing temporal tagging on historic documents and thus on documents written in Latin, for instance. Furthermore, some temporally annotated corpora have recently been made available for languages not supported by HeidelTime. For instance, the Portuguese and Romanian TimeBank corpora could be used to develop HeidelTime resources for these languages.

Temporal Tagging Literary Documents

As briefly mentioned in Section 3.3.5, in the context of the BMBF-funded heureCLÉA project, we are currently working on temporal phenomena in literary text (Bögel et al., 2014). For this, processing documents of the autonomic domain have to be improved. In addition, we have only tested and evaluated HeidelTime's quality on English scientific documents as representatives for the autonomic domain. We now address German literary text documents as another representative for this domain.

Automatic Language Detection

For a more convenient use of HeidelTime, the automatic identification of the language of the document that is to be processed would be helpful. Then, it would also be possible to run HeidelTime on a multilingual corpus without splitting the corpus into subsets for each language in advance. Although some language identification tools are publicly available, e.g., language identification based on TextCat (Cavnar and Trenkle, 1994) is part of the DKPro Core UIMA tool kit (Gurevych et al., 2007), we have not yet tested their integration in HeidelTime. Note that the language detection would have to be done before the linguistic preprocessing, i.e., outside HeidelTime's analysis engine in the UIMA version.

[51] In the meanwhile, we extended HeidelTime to cover temporal expressions referring to Historic Dates (Strötgen et al., 2014b).

[52] In the meanwhile, Chinese resources have been added to HeidelTime (Li et al., 2014) as well as Russian resources. This makes HeidelTime the first publicly available temporal tagger for Chinese and Russian temporal tagging.

Automatic Domain Detection

Since HeidelTime performs domain-sensitive temporal tagging, an automatic domain detection for the documents that are processed is conceivable. To detect the domain of a document, the results of our cross-domain corpus analysis could be used and formulated as features for a classification approach. Note, however, that some document types (documents from the news and colloquial domains), the document creation time (DCT) is important for the correct normalization of relative and underspecified expressions. Thus, to benefit from an automatic domain detection, it would be necessary to also tackle the task of automatic DCT detection. However, this is again a difficult task and highly depends on the source of the documents that are processed. For instance, popular news websites often contain a specific formatting which would have to be considered to identify the DCT.

3.8 Summary of the Chapter

In this chapter, we dealt with the topic of temporal tagging – in particular with multilingual and cross-domain temporal tagging. After having presented the state-of-the-art by surveying research competitions, temporally annotated corpora, and existing approaches to temporal tagging, we addressed the so far rarely considered issue of cross-domain temporal tagging by analyzing domain-dependent characteristics and suggesting strategies to address them. While most of the research on temporal tagging has been on processing English text documents so far, we focused on temporal tagging in a multilingual way, i.e., addressed additional languages.

Both issues – multilingual temporal tagging and cross-domain temporal tagging – are also considered by our newly developed temporal tagger HeidelTime. Since HeidelTime is built in a modular way with a strict separation between the source code and language-dependent resources, it can easily be extended to process further languages. This characteristic has been demonstrated since we developed HeidelTime resources for several languages but also by the fact that other researchers developed resources for two of the currently supported eight languages (Dutch and French). HeidelTime's high quality in both subtasks of temporal tagging, i.e., the extraction and the normalization of temporal tagging, has been demonstrated by describing HeidelTime's performance in two official research competitions, the results of a cross-domain evaluation, and – in general – by a wide range of evaluations of HeidelTime for several languages and domain settings. Finally, by making HeidelTime publicly available, we made important contributions to the research community.

4 The Concept of Spatio-temporal Events

In this chapter, we define the concept of an event, or more precisely of so-called spatio-temporal events. In addition to a concise definition, we describe how spatio-temporal events can be stored and organized, and how event instances can be compared with each other. Then, we also discuss how to extract spatio-temporal events from textual data. However, before doing so, we first motivate our approach of extracting spatio-temporal events and define the goals of this chapter.

4.1 Motivation and Objectives

Textual data ranging from corpora of digitized historic documents to large collections of news feeds and social media such as blogs provide a rich source of temporal and geographic information. Such types of information have recently gained a lot of interest in support of different search and exploration tasks. For instance, news documents can be organized along a timeline (e.g., Matthews et al., 2010) and documents can be placed on a map based on the documents' origin (e.g., Teitler et al., 2008). However, for this, temporal and geographic information embedded in documents is often considered in isolation.

If temporal information and geographic information were not considered in isolation but combined in a useful way, one could also extract simplistic versions of events because these are typically happening at some specific place at some specific time. Furthermore, we claim that for many categories of documents, events are essential to describe a topic or theme so that extracted event information can be used for manifold search and exploration tasks.

Thus, in this chapter, we address the following goals for exploiting combinations of temporal and geographic information extracted from documents: In Section 4.2, we survey the literature for different types of event definitions and concepts for being able to precisely define and narrow down in Section 4.3 the concept of an event as we will use it in our work. Since our event concept of so called *spatio-temporal events* is based on temporal and geographic information extracted from text documents, we also precisely define the two components of spatio-temporal events, i.e., extracted temporal and geographic expressions.

Then, in Section 4.4, we introduce and explain so-called temporal and geographic document profiles and how these concepts are extended to event document profiles. In Section 4.5, approaches to extract events from text documents are developed and evaluated.

In summary, in this chapter, we lay the foundations for developing search and exploration approaches exploiting spatio-temporal information extracted from documents. In addition, by combining geographic and temporal information extracted from documents, event-like features are built, which can be used in interesting and meaningful search and exploration tasks. These spatio-temporal and event-centric search and exploration approaches will be covered in Chapter 5 and Chapter 6, respectively.

4.2 Events in the Literature

The term *event* is used for many different things, both, in everyday life as well as in science. For instance, the Oxford Dictionary of English defines an event as follows:

> "a thing that happens or takes place, especially one of importance"
> (Soanes and Stevenson, 2003: p.600).

This definition is probably close to the everyday understanding of the term *event*, and also its etymological roots point into a similar direction.[1] Below, we will refer back to this definition when presenting other event concepts.

To get an idea of the diversity of event concepts, we also check Wikipedia's definition. When searching for "event" in Wikipedia, the user is referred to a disambiguation page[2] listing several meanings for event. In Table 4.1, we list for all pages linked on Wikipedia's disambiguation page under "events in science, technology, and mathematics" one sentence definitions for the respective concepts. While this table is intended to show the diversity of what can be understood as an event, in the following, we focus on event concepts that are relevant for our work, because the concept of an event exists – already in (related) science – in numerous fields with different understandings of what actually is an event.

4.2.1 Events in Philosophy

Already Table 4.1 shows that there is no universal definition for events in philosophy. Contrarily, there are several, sometimes conflicting theories about events. These conflicts can, generally speaking, partially be explained by different philosophical attitudes ranging from non-realism to realism and by the fact that even "*prima facie* commitments of human perception, action, language, and thought are not taken for granted in philosophy" (Casati and Varzi, 2010).

However, there are also multiple event concepts in favor of a realist attitude and "[w]hether [events] form a genuine metaphysical category is a question that has attracted the sustained interest of philosophers, especially in the second half of the 20th century" (Casati and Varzi, 2010). But there are still new essays and books written by philosophers to address the concept of events. In the following, we present examples of event definitions and concepts in philosophy.

Žižek's Journey on Events

One example of a recent philosophical work on events is the 2014 published book "Event: Philosophy in Transit" by the Hegelian philosopher Slavoj Žižek. While multiple examples of events are presented already at the beginning – "[a]n 'Event' can refer to a devastating natural disaster or to the latest celebrity scandal, the triumph of the people or a brutal political change, an intense experience of a work of art or an intimate decision" (Žižek, 2014: p.1) – the book is organized as a journey with each stop representing a putative event definition. Interestingly, already Žižek's first approximate definition – "an event is [...] the effect that seems to exceed its causes [...] [– goes straight to] the very heart of philosophy, since causality is one of the basic problems philosophy deals with" (Žižek, 2014: p.3).

[1] As described in the Oxford Dictionary of English, the etymological roots of *event* are "from Latin *eventus*, from *evenire* 'result, happen', from *e-*(variant of *ex*) 'out of' + *venire* 'come' " (Soanes and Stevenson, 2003: p.600).

[2] `http://en.wikipedia.org/wiki/Event` [last accessed June 10, 2014].

computing In computing, an event is an action or occurrence detected by the program that may be handled by the program.[2]
philosophy In philosophy, events are objects in time or instantiations of properties in objects. However, a universal definition has not been reached, as multiple theories exist concerning events.[3]
probability theory In probability theory, an event is a set of outcomes of an experiment (a subset of the sample space) to which a probability is assigned.[4]
relativity In physics, and in particular relativity, an event indicates a physical situation or occurrence, located at a specific point in space and time.[5]
synchronization primitive In computer science, an event (also called event semaphore) is a type of synchronization mechanism that is used to indicate to waiting processes when a particular condition has become true.[6]
UML An event in the Unified Modeling Language (UML) is a notable occurrence at a particular point in time.[7]
particle physics In particle physics, an event refers to the results just after a fundamental interaction took place between subatomic particles, occurring in a very short time span, at a well-localized region of space.[8]

Table 4.1: Wikipedia's "event" definitions in different fields of "science, technology, and mathematics" linked from Wikipedia's disambiguation page[1] for "event".
Sources: [1] `http://en.wikipedia.org/wiki/Event` *
[2] `http://en.wikipedia.org/wiki/Event_(computing)` *
[3] `http://en.wikipedia.org/wiki/Event_(philosophy)` *
[4] `http://en.wikipedia.org/wiki/Event_(probability_theory)` *
[5] `http://en.wikipedia.org/wiki/Event_(relativity)` *
[6] `http://en.wikipedia.org/wiki/Event_(synchronization_primitive)` *
[7] `http://en.wikipedia.org/wiki/Event_(UML)` *
[8] `http://en.wikipedia.org/wiki/Event_(particle_physics)` *
* [last accessed June 10, 2014].

While the Oxford dictionary definition for "event" includes the aspect of importance, the definitions presented by Žižek go even further, e.g., an event is considered as "a radical turning point, which is, in its true dimension, invisible" (Žižek, 2014: p.179). Furthermore, Žižek points out that "[t]here is, by definition, something 'miraculous' in an event" (Žižek, 2014: p.2) and that "an event at its purest and most minimal [is] something shocking, out of joint, that appears to happen all of a sudden and interrupts the usual flow of things" (Žižek, 2014: p.2).

However, there are also event theories in philosophy that do not require the importance factor to such an extent, e.g., events as they are defined in analytic philosophy and theories concerning the semantics of natural language. Before briefly describing examples of such theories below, we first present how events are distinguished from and "set [...] against entities belonging to other, philosophically more familiar, metaphysical categories" (Casati and Varzi, 2010) in the Stanford Encyclopedia of Philosophy.

Events in the Stanford Encyclopedia of Philosophy

While there is also a simple definition at the beginning of the entry to "event" in the Stanford Encyclopedia of Philosophy[3] – "[b]roadly understood, events are things that happen" (Casati and Varzi, 2010) – this definition is instantly criticized because the "broad characterization as 'things that happen' [...] clearly just shifts the burden to the task of clarifying the meaning of 'happen' " (Casati and Varzi, 2010). Thus, instead of trying to generally characterize *events*, they are compared to the metaphysical categories of *objects*, *facts*, *properties*, and *times* as we summarize in the following.[4]

- Events and objects: In contrast to events that occur, happen, or take place, material objects are said to exist. In addition, objects can move, are continuants, and persist through time, while events are occurrents and take up time. On the other hand, both, events and objects, are concrete, can be counted, compared, and referred to.
- Events and facts: While events are concrete and take up time, facts are abstract and characterized by the feature of a-temporality. Thus, it can be argued that for each event, there is an accompanying fact, namely that the event took place.
- Events and properties: While events are individuals and occur, properties are universal and recur. Nevertheless, there are also theories that not only events but also properties are spatio-temporally located.
- Events and times: In some theories, events are considered as properties of times, i.e., as times *cum description*, as temporal instants or intervals during which certain statements hold. More generally, events can be considered as spatio-temporal regions *cum description*.

For our work, the most important characteristic of events described in the Stanford Encyclopedia of Philosophy is that events occur in space and time and can thus be spatio-temporally located.

Event Theories

In the following, we briefly present the characteristics of two event theories, which are – according to the Internet Encyclopedia of Philosophy[5] – among "the leading theories of events" (Schneider, 2014), namely those of Jaegwon Kim and Donald Davidson. In general, the goal of event theories is "to propose and defend an identity condition on events" (Schneider, 2014), i.e., to determine when two events are equal, although the theories do not have to be restricted to this goal.

For Kim, events are instantiations of properties and structured with an event being defined as a three-tuple of an object x (or a set of objects x_n), a property P, and a time or time interval t. He defines the two basic principles of existence and identity as follows: First, the existence principle states that "[(x_n, t), P] exists if and only if the n-tuple of concrete objects x_n exemplifies the n-adic empirical attribute P at time t" (Kim, 1973: p.223). Then, the identity principle is first defined for monadic events, i.e., events concerning only one object as "[(x, t), P] = [(y, t'), Q] if and only if x=y, t=t', and P=Q." (Kim, 1973:

[3] The Stanford Encyclopedia of Philosophy, `http://plato.stanford.edu/index.html` [last accessed June 25, 2014].

[4] Since this itemization is a summary of the comparison of events to other categories described on the "event" page of the Stanford Encyclopedia of Philosophy(Casati and Varzi, 2010), we omit citations and refer to that page and the numerous references mentioned therein.

[5] The Internet Encyclopedia of Philosophy, `http://www.iep.utm.edu/` [last accessed June 25, 2014].

p.223). This identity principle is later generalized to n-adic events. Although not explicitly stated in the definitions, for two events to be identical, they do not only have to occur at the same time but also at the same location since objects involved in events exist at particular times at specific locations: "[...] these are indeed different events. Consider, for example, their locations: the first obviously took place [...] [at location 1], but it is not clear where the second event occurred, [...] but clearly not [at location 1]" (Kim, 1973: p.224).

Grounded on these definitions of the two principles, according to Kim, "events are non-repeatable, concrete particulars, including not only changes but also states and conditions, [...] [e]ach event has a spatiotemporal location, [...] and [a]lthough events may exemplify any number of properties, only one property, the constitutive property, individuates the event" (Schneider, 2014). Thus, Kim's events are quite fine-grained.

Donald Davidson, "one of a handful of philosophers in the latter half of the twentieth century who reshaped the terrain of analytic philosophy [...] [studying the] nature of linguistic meaning" (Lepore and Ludwig, 2013: p.1), addresses the questions when events are identical, when distinct, and "[w]hat criteria are there for deciding one way or the other in particular cases" (Davidson, 1969). In his early work, Davidson sets up the identity criterion as "events are identical if and only if they have exactly the same causes and effects" (Davidson, 1969). However, due to circularity issues – "what is a cause or effect [...] if not an event?" (Schneider, 2014) – he later sets up a second identity criterion following suggestions of W. V. A. Quine: events are identical if they occur in the same place at the same time. Events can thus also be distinguished from objects because "events occur at a time in a place while objects occupy places at times" (Davidson, 1985).

In summary, according to Davidson, events "are spatiotemporally individuated entities, standing in part-whole relationships, as well as in causal relations to other events" (Stoecker, 2013: p.16). As for Kim, Davidson's events are "particular, non-repeatable occurrences" (Schneider, 2014) while they are less fine-grained than Kim's events.

Summary

While we presented some examples of event concepts in philosophy, this overview is of course far from complete. Note, however, that even Žižek states about his new, more than 200 pages long book on events that his "overview is, of course, far from complete" (Žižek, 2014: p.208). Depending on the point of view, the importance factor which makes something an event is quite different. While Žižek's event definitions set a high value on this aspect, the presented philosophical event theories of Kim and Davidson do not account for the importance factor to such an extent.

For our work, the most important characteristics of events in philosophy are that they are often considered as particulars, that they can be counted and referred to, and that they can be spatio-temporally located. Of course, there are many further event theories in philosophy, in which the spatio-temporal criterion is of uttermost importance, e.g., David Lewis' event theory (see, e.g., Casati and Varzi, 2010; Schneider, 2014). However, we do not aim at providing a complete overview of events in philosophy and thus switch to events in linguistics in the following.

4.2.2 Events in Linguistics

Another field of research studying concepts of events is linguistics. Obviously, there is a partial overlap between the different research fields, e.g., the concepts already described in the previous section – in particular the event theories of Kim and Davidson – could also be listed under "events in linguistics". This is due to the fact that the respective philosophers, and analytic philosophers in general, deal with the meaning of natural language. The following statement about Gottlob Frege – often considered as "the father of analytical philosophy" (Dummett, 2007) – in the Oxford Handbook of The History of Analytic Philosophy underlines this fact: "Beginning with work by Grice, Strawson, and Austin [...] and exploding in the work of [...] Davidson [...] and others, the use of Fregean ideas in understanding the semantics and logical form of natural language became a central area of philosophy. Some of this work fed into theories of meaning and reference in linguistics" (Burge, 2013: p.364). In other words, one area in linguistics studying events is logical semantics.

However, in general, there are different areas of linguistics that deal with events, namely lexical semantics, logical semantics, and syntax. Lexical semantics generally studies word meaning, but when addressing events, "lexical semanticists must look outward from the verb to the sentence in order to characterize the effects of a verb's event structure" (Tenny and Pustejovsky, 2000a). In contrast, logical semantics generally deals with "compositional properties of propositional interpretations" (Tenny and Pustejovsky, 2000a), but when events come into play "logical semanticists must look inward from the sentence to the verb to represent semantic facts that depend on event-related properties of particular verbs" (Tenny and Pustejovsky, 2000a). Finally, the interactions between syntactic structures and semantics of events is addressed in research on syntax. The common denominator of these research aspects is to study events as *grammatical objects*.

It is important to mention that events as subjects of analysis in linguistics are referred to "as grammatically or linguistically represented objects, not as events in the real world" (Tenny and Pustejovsky, 2000a). But even when considering events as grammatical objects, there are two research questions, namely (i) "whether the grammar of natural language [...] represent[s] events in some way, apart from any internal structure of that event, [...] [and (ii)] whether 'grammaticalized' events have any internal structure which is also grammaticalized" (Tenny and Pustejovsky, 2000a).

However, since in our work we are interested in the events in the world described in natural language texts, we do not present any further details about events as grammatical objects in linguistics, but refer to the book "Events as Grammatical Objects" (Tenny and Pustejovsky, 2000b) in which the above mentioned research questions are studied, and to the introduction of that book giving an overview of the history of events in linguistic theory (Tenny and Pustejovsky, 2000a). Despite that, we will present two well-known (computational) linguistic event concepts in Section 4.2.4, after detailing to what the term "event" refers in several research areas of computer science and natural language processing.

4.2.3 Events in Computer Science and Natural Language Processing

The term *event* is not only used in event theories and concepts developed in philosophy and linguistics, but also in several computer science and natural language processing research fields. In the following, we present example research areas in which event concepts have been used and defined. The main intention of this overview is to avoid confusion between existing event concepts and our event concept which will be developed in Section 4.3. Our focus will be on analyzing the importance of the temporal and spatial characteristics of the presented event concepts, and on how the event data is accessible.

Events in Visual Motion Analysis

A computer science research area in which the term *event* is used to refer to an important concept is visual motion analysis (VMA). A goal in VMA is to analyze and interpret video data by tracking movements of so-called interest points or objects. In general, interest points are local image features. Laptev (2005) introduces the concept of spatio-temporal interest points and shows "that the resulting local space-time features often correspond to interesting events in video data" (Laptev, 2005). For the detection of so-called "spatio-temporal events, [...] local structures in space-time [are detected] where the image values have significant local variations in both space and time" (Laptev, 2005). Examples of spatio-temporal events, which can be detected by Laptev's algorithm are amongst others sequences of walking persons and hand-waving gestures.

In contrast to other types of events that will be described in the following, events in VMA are not to be detected in textual data but in sequences of images. However, their description and in particular their naming as *spatio-temporal events* show how important temporal and spatial characteristics of events are.

Events in Community Detection

Community detection in graphs is not only important in computer science but in several disciplines such as sociology and biology, i.e., in all "disciplines where systems are often represented as graphs" (Fortunato, 2010). In general, communities "can be considered as fairly independent compartments of a graph" (Fortunato, 2010), and their detection is helpful for all kinds of analyses because vertices being part of the same community typically share some common characteristics. A lot of research on community detection is assuming static graphs although "[t]he study of the evolution of these graphs over time can provide tremendous insight on the behavior of entities, communities and the flow of information among them" (Asur et al., 2007). Once the evolutionary behavior is studied, events come into play.

For characterizing the dynamic behavior of interaction graphs, Asur et al. (2007) define several types of events involving communities (continue, k-merge, k-split, form, and dissolve) and events involving individuals (appear, disappear, join, and leave). While these events are used "to characterize complex behavioral patterns of individuals and communities over time" (Asur et al., 2007), it is obvious that all these events have a clear temporal component since they occur and are detected between two snapshots of the analyzed interaction graph. In addition, the events affect either communities or entities at some specific positions in the graph so that the events in community detection also have a spatial component. However, as in visual motion analysis, natural language processing is not involved since the events occur in graphs and not in textual data.

Events in Social Networks

While social network analysis is one of the research fields in which community detection is important, there is usually another kind of event definition than the general community detection event types described above because "an instance of cooccurring actors is termed an *event* in social network terms" (Lauw et al., 2005). Using so-called affiliation networks, i.e., networks with a set of actors, a set of events, and links between actors and events, it is then possible to infer "association[s] between actors through their participation in events" (Lauw et al., 2005).

In addition to the general social network event definition, Lauw et al. (2005) also define so-called *spatio-temporal events* since they present an approach to mine social networks from spatio-temporal data

such as "physical locations of moving objects, or [...] cyber locations visited by Internet users" (Lauw et al., 2005). Each unit of such data sets consists of location and time information and is associated to a specific individual. In their approach, a spatio-temporal event occurs if, among other constraints, there are at least two units with identical location information and with identical time information (within a tolerance interval), and if there are at least two actors involved.

As for several other event concepts surveyed in this section, the temporal and spatial components are fundamental for the events defined by Lauw et al. (2005). Thus, their event concept is quite related to the one we will define in Section 4.3. However, in their work, natural language processing is not required at all since their events can be directly inferred from the spatio-temporal data.

Events in Sensor Networks

"Sensor networks are distributed event-based systems" (Krishnamachari et al., 2002) and are applied in many areas such as disaster monitoring and object tracking. An important task in sensor network research is event detection, "which aims at identifying emergent physical phenomena and make real-time decisions about physical environments" (Yin et al., 2009). As for several other event concepts and definitions, spatial and temporal information is a crucial aspect of events in sensor networks. "[S]ensor nodes are deployed in a physical space, [...] sensor readings are collected over a period of time, [...] [and] the changes in sensor readings caused by an event usually exhibit strong spatio-temporal correlations" (Yin et al., 2009). Thus, events in sensor networks are often named *spatio-temporal events*. However, while the temporal and spatial characteristics of events in sensor network research are crucial, spatio-temporal events in this context are typically not extracted form textual data so that natural language processing is not involved.

In contrast to such sensor networks, there are also applications and research approaches relying on the "humans as sensors" concept. For instance, in the RESCUE (Responding to Crises and Unexpected Events) project, one of the goals is "extracting meaningful events from multimodal data streams" (Mehrotra et al., 2004) including textual data such as transcribed eyewitness interviews. The event extraction – with events defined as "significant phenomenon or occurrence embedded in space and time" (Mehrotra et al., 2004) – is performed based on a dynamic taxonomy of crisis events. Each event type is associated with a set of properties, and the goals are to extract these properties as well as to spatio-temporally locate the events. Note, that in particular the supplemented information about location and time, which is available as meta information to the texts, is helpful for the latter task.

Events in Biomedical NLP

In biomedical natural language processing (BioNLP), the extraction of events from textual data plays a crucial role. In this context, events are either some kind of interactions between genes, proteins, or other biomedical entities, e.g., protein-protein interactions as in the BioCreative challenges (e.g., Krallinger et al., 2008), or more complex behavior of bio-molecules as in the BioNLP shared tasks on event extraction (e.g., Kim et al., 2009). Thus, while biomedical events also take place at specific times and places, e.g., within species or cells, the goals of event extraction in BioNLP are to determine all participating entities and the type of event. The latter task is usually to select an event type in a predefined event ontology.

Unlike other event concepts, biomedical events are not very similar to the events we will define in the next section. While extracted from textual data, the temporal and spatial aspects are of minor importance.

Events in Topic Detection and Tracking

The idea behind topic detection and tracking (TDT) is an "event-based organization of broadcast news [...] [with the goal of clustering all] news stories that are strongly related by some seminal real-world event" (Allan, 2002a). Note that an event in the context of TDT is generally defined as "something that happens at some specific time and place" (Allan, 2002a), but the goal of TDT is not to build a cluster for each event but for each topic. A topic is defined as "a seminal event or activity, along with all directly related events and activities" (Allan, 2002a). Thus, main tasks in TDT are (i) to recognize "the onset of a new topic in the stream of news stories" (Allan, 2002a) and (ii) to add all news stories that are related by the same seminal event.

In summary, while the two key components of an event in TDT-related works are the temporal component and the geographic component, there is a distinction between an event in general and a "seminal real-world event" that is important enough to start a new topic. The focus of TDT lies on detecting the latter types of events.

Events in Recommender Systems

Real-world events have also been addressed in the context of recommender systems. For instance, Khrouf and Troncy (2013) describe events as "a natural way for referring to any observable occurrence grouping persons, places, times and activities" (Khrouf and Troncy, 2013). Assuming that information about events is available in structured format, i.e., no natural language processing is required in this approach, they build an event model in terms of what, where, and when an event does happen as well as who is involved. Based on this model, a system is developed for event recommendation exploiting information about "user preferences (ratings, likes, etc.), the attended events (visited places, involved artists) [...] [and] restrictions such as time, location, category, popularity and which friend will attend" (Khrouf and Troncy, 2013).

Obviously, the temporal and geographic aspects of events are crucial for event recommendation since only upcoming events should be recommended and the geographic closeness has a high influence on whether a user may be interested in attending an event.

Summary

In many of the event concepts presented in this section, the temporal and the spatial components play a crucial role. Nevertheless, there are of course significant differences between the event concepts. In addition, this overview of events in computer science and natural language processing is obviously far from complete. While in some research areas, occurring event concepts are not that important, other event concepts are not very related to our work. However, there are also two event concepts which have not yet been presented although they are highly related to our work. However, due to their importance, and in particular because we already introduced the contexts of these event concepts in Chapter 3, we allocate a separate section to these event concepts.

4.2.4 ACE and TimeML Events

The Automated Content Extraction (ACE) program and the temporal markup language TimeML have already been introduced in Chapter 3 when presenting temporally annotated corpora, research challenges, as well as annotation standards and specifications for temporal expressions. However, in both contexts, not only temporal expressions but also *events* are fundamental elements.

Events in ACE

The Automatic Content Extraction (ACE) program was started in 1999 and aimed at developing "technology to automatically infer from human language data the entities being mentioned, the relations among these entities that are directly expressed, and the events in which these entities participate" (Doddington et al., 2004). The central concept in ACE are the entities, and the research objectives are defined in such a way that identifying the entities themselves is crucial and not just the identification of terms referring to any entity. Thus, instead of named entity recognition, named entity normalization and coreference resolution are of major interest in ACE.

Due to the importance of entities, events in ACE are also "represented in terms of their attributes and their participants [...] and each participant is characterized by a role that it plays in the event (agent, object, source, target)" (Doddington et al., 2004). There are eight event types with a total of 33 subtypes, e.g., the main event type "Life" contains subtypes such as "Be-Born" and "Marry".[6] Using the ACE XML-based markup language, the goal is to "tag the textual mention or anchor for each event, [...] categorize it by type and subtype, [...] [and] identify event participants [...] and attributes [...] according to a type-specific template" (Doddington et al., 2004).

The attributes of ACE events can be of different forms, e.g., temporal, locative as well as others like instrument or purpose, but although events are defined in terms of their attributes and participants, they do not have required key components. For instance, neither a temporal anchoring nor a geographic grounding is obligatory. In general, "arguments will be taggable only when they occur within the scope of the corresponding event, typically the same sentence" (Liao and Grishman, 2010). However, since each event template contains slots for temporal and geographic information, ACE events also occur at specific times and places although these may not be mentioned explicitly in a text.

In summary, the extraction of ACE events is quite similar to the task of semantic role labeling, which can be broadly defined as "who did what to whom, and when, where and how?" (Palmer and Xue, 2010: p.246). Note, however, that while the 33 event subtypes occur frequently in news articles, ACE events are limited to these predefined event types. Thus, ACE events do not cover other types of events although they might appear in a text. This is an important difference to TimeML events described next.

Events in TimeML

TimeML is an XML-based markup language for temporal annotation. Its goal is "to capture the richness of temporal and event related information in text [...] [by providing] a language for the representation of temporal relations" (Pustejovsky et al., 2005). TimeML was developed in the research context of question answering systems and its creation is motivated by the facts that "[e]vents in articles are naturally anchored in time within the narrative of a text [...] [and that] temporally grounded events are the very foundation from which we reason about how the world changes" (Pustejovsky et al., 2003a). Thus, an essential step for information extraction and applications relying on information extraction, e.g., question answering and summarization systems, is "to identify what events are being described and to make explicit when these events occurred" (Pustejovsky et al., 2005).

Since "TimeML separates the representation of event and temporal expressions from the anchoring or ordering dependencies that may exist in a given text" (Pustejovsky et al., 2003a), there are four major data

[6] For a complete overview of event types and subtypes, we refer to "The ACE 2005 (ACE05) Evaluation Plan" description: `http://www.itl.nist.gov/iad/mig/tests/ace/2005/doc/ace05-evalplan.v3.pdf` [last accessed July 7, 2014].

structures in TimeML: events (EVENT), temporal expressions (TIMEX3), temporal signals (SIGNAL), and relations (LINK). While we already introduced TimeML and its TIMEX3 tag in Section 3.2.1, temporal signals are function words that are important to determine temporal relations between times and events, relations are, for instance, the temporal relations defined in Allen (1983)'s interval algebra (cf. Section 2.3.1) that hold between times and events, and events are the concept we are focusing on in the following.

The concept *event* is used in TimeML as "a cover term for situations that happen or occur [...] [and for] predicates describing states or circumstances in which something obtains or holds true" (Pustejovsky et al., 2003a). The main aspect of events in TimeML is therefore the temporal dimension, i.e., that events can be temporally anchored and ordered. Consequently, events in TimeML are also sometimes referred to as *temporal events*.

There are several realizations of events, namely tensed or untensed verbs, nominalizations, adjectives, predicative clauses, and prepositional phrases (Saurí et al., 2006). An event can either be "punctual or last for a period of time" (Pustejovsky et al., 2003a) and is of one of the following classes: reporting, perception, aspectual, intensional action, intensional state, state, or occurrence. Note that the main requirement that something is marked as an event is that it can be anchored in time. Thus, generics are not considered as markable events in TimeML since generic formulations "do not explicitly refer to specific events" (Saurí et al., 2006). However, in contrast to ACE events, TimeML events are not limited to a predefined set of event types.

In summary, events in TimeML contain only one key component, namely the temporal dimension. In addition, there is no requirement that an event is of a particular importance. It is only necessary that no generic circumstances are described but either explicit instances of events or states that may change over time. Although, similar to TimeML, SpatialML (Mani et al., 2008) was developed as a markup language for spatial information occurring in text documents, TimeML events are not (yet) linked to geographic information of SpatialML annotations.

4.2.5 Further Event Types and Summary

While we already presented a broad variety of *event* concepts in the previous sections, there are of course further event definitions in the literature. In addition, there are also differently named concepts which are similar to our concept of spatio-temporal events initially introduced in (Strötgen and Gertz, 2010b; Strötgen et al., 2011; Strötgen and Gertz, 2012a). Such similar concepts, e.g., spatio-temporal facts introduced by (Wang et al., 2011), will be considered in Chapter 6 where we present related approaches about searching and exploring events extracted from text documents. In the next section, however, we define and explain our concept of spatio-temporal events.

4.3 The Concept of Spatio-temporal Events

In contrast to the partially complex definitions and characteristics of events surveyed in the previous section, we define an event in a quite simplistic way. Based on the assumption that events often take place at some specific time and at some specific place, we assume that many events in textual documents are described using temporal and geographic expressions. Since our events are based on temporal and geographic information, we name our concept of an event a *spatio-temporal event*.

4.3.1 Definition

Generally speaking, we define an event in the following way: if a geographic and a temporal expression cooccur in a textual document – or within a specific window in the document (e.g., in a sentence) – the temporal expression and the geographic expression form an event if the text describes something happening at that specific time and place. More formally, we define a *spatio-temporal event* as follows:

Definition 4.1. *(Spatio-temporal Event)*
Given a document d and extracted temporal and geographic expressions te_i and ge_j, respectively. The tuple $\langle te_i, ge_j \rangle$ forms a *spatio-temporal event* if te_i and ge_j cooccur within a specific window w in d, and if the text describes something happening at the time referred to by te_i at the place referred to by ge_j.

Despite its simplicity, our event concept has several advantages for complex search and exploration tasks. Since both components of spatio-temporal events are well defined, can be normalized, and can be organized hierarchically (cf. Section 2.3.1 and Section 2.4.1), the same characteristics hold for spatio-temporal events. Thus, they are term- and language-independent, they are comparable with each other, and they can be validated against query intervals and regions when searching for events. These characteristics will be of uttermost importance for the search and exploration scenarios which will be developed in Chapter 6.

Typically, the temporal expressions being part of spatio-temporal events are of the types "date" or "time", but also "duration" expressions can sometimes be anchored on a timeline (cf. Section 2.3.2) so that these can also be part of spatio-temporal events. Note that depending on the temporal and geographic expressions forming a spatio-temporal event, both components of an event can be of different granularities. For instance, the event ⟨"March 11, 2013", "Heidelberg"⟩ is quite fine-granular since the geographic expression is of type "city" and the temporal expression is of granularity "day". In contrast, the event ⟨"2013", "Germany"⟩ is a quite coarse event with the geographic expression being of type "country" and the temporal expression of granularity "year".

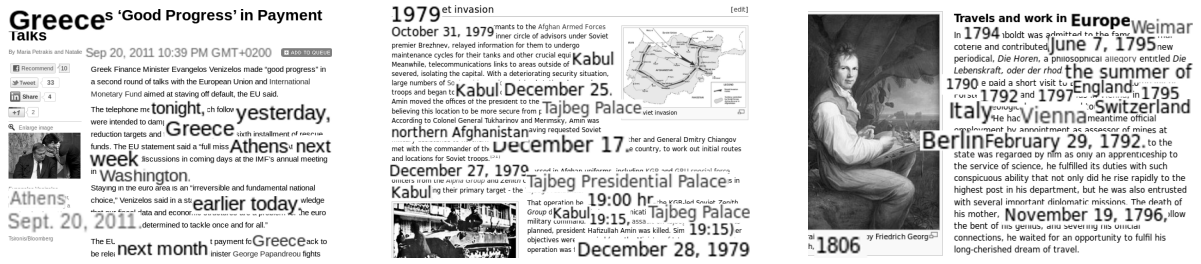
Details about the different granularities of temporal and geographic expressions as well as about what kind of information is carried by extracted temporal and geographic expressions will be explained in Section 4.3.3 when defining the concepts of extracted temporal and geographic expressions. However, for a better understanding of the concept of spatio-temporal events, we first discuss several examples.

4.3.2 Examples of Potential Spatio-temporal Events

In Figure 4.1, the three example documents introduced in Chapter 3 (Figure 3.1) are depicted again, showing now that not only temporal but also geographic information is quite frequent in many types of documents. In the following, we analyze some potential spatio-temporal events described in these documents assuming that the window size w is set to one sentence.

The first two potential spatio-temporal events under analysis are extracted from the following sentence of the news document shown in Figure 4.1(a).

Spatio-temporal Event Example – Sentence 1.
Evangelos Venizelos, <PLACE>*Greece*</PLACE>*'s finance minister, listens to an aide during a session of parliament in* <PLACE>*Athens*</PLACE> *on* <TIME>*Sept. 20, 2011*</TIME>.
potential event e_1 = ⟨"Sept. 20, 2011", "Greece"⟩
potential event e_2 = ⟨"Sept. 20, 2011", "Athens"⟩ → *valid spatio-temporal event*



(a) News document. (b) History document. (c) Biography document.

Figure 4.1: Examples of three types of documents, in which temporal and geographic information occurs frequently.
Sources: (a): `http://www.bloomberg.com/`.
(b): `http://en.wikipedia.org/wiki/Soviet_war_in_Afghanistan`.
(c): `http://en.wikipedia.org/wiki/Alexander_von_Humboldt`.

Although the temporal expression "Sept. 20, 2011" and the geographic expression "Greece" cooccur in the predefined window of one sentence, the first potential event $e_1 = \langle$"Sept. 20, 2011", "Greece"$\rangle$ is not a valid spatio-temporal event since the extracted place information "Greece" is not used to refer to the location where an event described in the sentence takes place. In contrast, the second potential event $e_2 = \langle$"Sept. 20, 2011", "Athens"$\rangle$ is a valid spatio-temporal event since the sentence describes something happening in "Athens" on "Sept. 20, 2011". That is, the temporal expression "Sept. 20, 2011" and the geographic expression "Athens" do not only cooccur in w but they also refer to something happening at the referred time and place.

The following sentence from the same news document contains a total of three geographic expressions and two temporal expressions so that there are six potential spatio-temporal events.

Spatio-temporal Event Example – Sentence 2.

The <PLACE>*EU*</PLACE> *statement said a "full mission" will return to* <PLACE>*Athens*</PLACE> <TIME>*next week*</TIME> *after his discussion in* <TIME>*the coming days*</TIME> *at the IMF's annual meeting in* <PLACE>*Washington*</PLACE>.

potential event e_3 = $\langle$"next week", "EU"$\rangle$
potential event e_4 = $\langle$"next week", "Athens"$\rangle$ $\rightarrow$ *valid spatio-temporal event*
potential event e_5 = $\langle$"next week", "Washington"$\rangle$
potential event e_6 = $\langle$"the coming days", "EU"$\rangle$
potential event e_7 = $\langle$"the coming days", "Athens"$\rangle$
potential event e_8 = $\langle$"the coming days", "Washington"$\rangle$ $\rightarrow$ *valid spatio-temporal event*

Since the two temporal expressions and the three geographic expressions cooccur within the same sentence, the first condition for valid spatio-temporal events is satisfied by all potential events. However, only e_4 and e_8 are valid spatio-temporal events because there is something described in the sentence happening in "Athens" "next week" and in "Washington" "the coming days". In the same way as the geographic expression "Greece" in the previous example, the geographic expression "EU" is again not used to refer to a location where something is happening but as a modifier of "statement". In addition, "next week" and "Washington" as well as "the coming days" and "Athens" do not syntactically belong together.

Similar to the examples extracted from the news document, there are also several potential events in the history document depicted in Figure 4.1(b). The first sentence under analysis contains one temporal expression and two geographic expression, i.e., two potential spatio-temporal events:

Spatio-temporal Event Example – Sentence 3.
On <TIME>*December 27, 1979* </TIME>*, 700 Soviet troops dressed in Afghan uniforms, including KGB and GRU special force officers from the Alpha Group and Zenith Group, occupied major governmental, military and media buildings in* <PLACE>*Kabul* </PLACE>*, including their primary target - the* <PLACE>*Tajbeg Presidential Palace* </PLACE>*.*
potential event e_9 = ⟨"December 27, 1979", "Kabul"⟩ → *valid spatio-temporal event*
potential event e_{10} = ⟨"December 27, 1979", "Tajbeg Presidential Palace"⟩ → *valid spatio-temporal event*

Both potential events are valid spatio-temporal events because it is described how something is happening in "Kabul" and at the "Tajbeg Presidential Palace". Although one might argue that it is only one event happening at one location because the "Tajbeg Presidential Palace" lies in "Kabul" or that it is the same event happening at two locations – one location of coarser granularity and one location of finer granularity – the explicit usage of two geographic expressions results in two valid spatio-temporal events.

In the following two example sentences from the historic document, there is only one potential event in each sentence. Obviously, in both sentences something is happening at the locations referred to by the geographic expressions ("Kabul" and "Tajbeg Palace", respectively) at the times referred to by the temporal expressions ("December 25" and "19:15", respectively). Thus, both events are valid spatio-temporal events.

Spatio-temporal Event Example – Sentence 4.
With a deteriorating security situation, large numbers of Soviet airborne forces joined stationed ground troops and began to land in <PLACE>*Kabul* </PLACE> *on* <TIME>*December 25* </TIME>*.*
potential event e_{11} = ⟨"December 25", "Kabul"⟩ → *valid spatio-temporal event*

Spatio-temporal Event Example – Sentence 5.
At <TIME> *19:15* </TIME>*, the assault on* <PLACE>*Tajbeg Palace* </PLACE> *began; as planned president Hafizullah Amin was killed.*
potential event e_{12} = ⟨"19:15", "Tajbeg Palace"⟩ → *valid spatio-temporal event*

Finally, there are also several temporal and geographic expressions in the biography document shown in Figure 4.1(c). The following two example sentences contain one potential event each. Again, both potential events are valid spatio-temporal events since the sentences describe something happening in "England" and "Berlin" in "the summer of 1790" and on "February 29, 1792", respectively.

Spatio-temporal Event Example – Sentence 6.
In <TIME>*the summer of 1790* </TIME> *he paid a short visit to* <PLACE>*England* </PLACE> *in the company of Forster.*
potential event e_{13} = ⟨"the summer of 1790", "England"⟩ → *valid spatio-temporal event*

Spatio-temporal Event Example – Sentence 7.
He had obtained in the meanwhile official employment by appointment as assessor of mines at <PLACE>*Berlin* </PLACE>*,* <TIME>*February 29, 1792* </TIME>*.*
potential event e_{14} = ⟨"February 29, 1792", "Berlin"⟩ → *valid spatio-temporal event*

However, there are also more complex sentences with several potential events due to a rather large number of temporal and geographic expressions, as in the following example from the same bibliography document:

Spatio-temporal Event Example – Sentence 8.
In <TIME>*1792* </TIME> *and* <TIME>*1797* </TIME> *he was in* <PLACE>*Vienna* </PLACE>*; in* <TIME>*1795* </TIME> *he made a geological and botanical tour through* <PLACE>*Switzerland* </PLACE> *and* <PLACE>*Italy* </PLACE>*.*
potential event e_{15} = ⟨"1792", "Vienna"⟩ → *valid spatio-temporal event*
potential event e_{16} = ⟨"1792", "Switzerland"⟩
potential event e_{17} = ⟨"1792", "Italy"⟩
potential event e_{18} = ⟨"1797", "Vienna"⟩ → *valid spatio-temporal event*
potential event e_{19} = ⟨"1797", "Switzerland"⟩
potential event e_{20} = ⟨"1797", "Italy"⟩
potential event e_{21} = ⟨"1795", "Vienna"⟩
potential event e_{22} = ⟨"1795", "Switzerland"⟩ → *valid spatio-temporal event*
potential event e_{23} = ⟨"1795", "Italy"⟩ → *valid spatio-temporal event*

Due to the three temporal and three geographic expressions, there is a total of nine potential events. While the temporal expressions "1792" and "1797" form valid events with the geographic expression "Vienna", only "1795" forms valid events with the geographic expressions "Switzerland" and "Italy". Thus, only four of the nine potential events are valid spatio-temporal events.

Reasons for Cooccurring Expressions not Forming Events

There are different reasons why some of the potential events are not valid events, e.g., the temporal and geographic expressions do not syntactically belong together, or the geographic expression is not used to refer to a location but is attributively used such as in the case of the potential event e_1 (Evangelos Venizelos, <PLACE>Greece</PLACE>'s finance minister, ...) in the *Spatio-temporal Event Example – Sentence 1.*

In Section 4.5, different approaches to extract spatio-temporal events from text documents will be presented. There, we will refer back to the examples of potential spatio-temporal events discussed above, and we will also analyze further examples to identify challenges for spatio-temporal event extraction.

4.3.3 Normalized Spatio-temporal Events

Note that the temporal and geographic components of the spatio-temporal events discussed above have not been normalized, and only the text strings have been used in the examples for the sake of readability. Obviously, the temporal and geographic components are rather less valuable for search and exploration tasks if only the pure text strings occurring in the documents' texts were extracted. As already explained in Section 2.3.1 and Section 2.4.1, temporal and geographic expressions can be normalized and organized hierarchically and this information will be part of extracted spatio-temporal events. Thus, in this section, we concisely define *extracted temporal and geographic expressions.* In Section 4.4, we further explain how extracted temporal and geographic expressions can be organized in temporal and geographic document profiles so that their semantics can be exploited for spatio-temporal search and exploration tasks. To exploit the semantics of valid combinations of temporal and geographic expressions, i.e., of spatio-temporal events, we introduce event document profiles, additionally.

Defining Extracted Temporal and Geographic Expressions

As briefly explained for geographic expressions (cf. Section 2.4.3) and in detail for temporal expressions (cf. Chapter 3), there are so-called geo-taggers and temporal taggers that do not only mark or extract geographic and temporal expressions, respectively, but that also assign some normalized information to

the extracted expressions. For many applications relying on extracted temporal and geographic information, it is important to know what the temporal and geographic expressions mean, i.e., to what location or point in time they refer. For instance, in the *Spatio-temporal Event Example – Sentence 2*, the temporal expressions "next week" and "the coming days" can refer to any points in time depending on the reference time. In addition, the geographic expressions "Athens" and "Washington" can refer to many different locations. For instance, Athens can refer to locations in Greece, Ohio, or Georgia (the one in the U.S. not the country in the Caucasus region in Eurasia) and Washington to Washington, D.C. or Washington state – to just name a few. Thus, it is crucial that normalization information is associated to each extracted temporal and geographic expression – and thus to each spatio-temporal event – so that we define an *extracted temporal expression* and an *extracted geographic expression* as follows:

Definition 4.2. *(Extracted Temporal Expression)*
Given a document d, an *extracted temporal expression* te_i is a three tuple $te_i = \langle t_i, s(t)_i, p(t)_i \rangle$, with t_i being the temporal expression, $s(t)_i$ the normalized semantics of t_i, and $p(t)_i$ the document and offset information, i.e., the start and end position in d.

Definition 4.3. *(Extracted Geographic Expression)*
Given a document d, an *extracted geographic expression* ge_i is a three tuple $ge_i = \langle g_i, s(g)_i, p(g)_i \rangle$, with g_i being the geographic expression, $s(g)_i$ the normalized semantics of g_i, and $p(g)_i$ the document and offset information, i.e., the start and end position in d.

In the case of extracted temporal expressions, the semantics feature $s(t)$ contains all the normalization information that is assigned to the expression by the temporal tagger as it was described in Chapter 3. For example, the type information of the temporal expression (date, time, duration, or set), the normalized value information as most important attribute, and all further attributes containing normalization information are contained in the semantics feature $s(t)$ of an extracted temporal expression. Note that based on the value information of time and date expressions, their granularity can directly be determined, and is thus provided together as part of the type information, e.g., as "date-year" or "date-day".

Analogously, the semantics feature $s(g)$ of extracted geographic expressions contains all the information about the location assigned to the expression by the geo-tagger. As described in Section 2.4.3, a geo-tagger usually assigns to each location a unique identifier of an entry in a gazetteer. Thus, $s(g)$ contains, the identifier itself, but also type information (e.g., city, state, country) and latitude/longitude information. In addition, containment information can also be directly accessed using the gazetteer.

To provide an example of an extracted temporal expression and an extracted geographic expression, we use the temporal and geographic expressions forming the event e_2 occurring in the *Spatio-temporal Event Example – Sentence 1* (e_2 = ⟨"Sept. 20, 2011", "Athens"⟩). Assuming that (i) the geo-tagger extracts "Athens" as the capital of Greece, (ii) the geo-tagger's gazetteer relies on GeoNames, (iii) the temporal tagger correctly extracts and normalizes "Sept. 20, 2011", (iv) the ID of the news document is "news-1" and that (v) the temporal and geographic expressions are the first and second occurring temporal and geographic expressions, respectively, then the extracted temporal and geographic expressions are:

te_1 = ⟨"Sept. 20, 2011", [tpye:date-day, value:2011-09-20], [docid:news-1, start:231, end:244]⟩

ge_2 = ⟨"Athens", [id:264371, type:city, lat/long:37.98/23.72, parent:["Attica", id:6692632]], [docid:news-1, start:221, end:226]⟩

Assuming that all temporal and geographic expressions occurring in text documents are extracted and normalized in such a way using a temporal and a geo-tagger, the extracted temporal and geographic expressions can be organized in so-called document profiles as described in the following.

4.4 Document Profiles

In general, the idea behind document profiles is to describe and organize important information that is associated with a document in a concise and easy-accessible manner before such information is further utilized. For instance, Alonso (2008) introduced temporal document profiles to make explicit all date expressions mentioned in a document no matter how they occurred (explicitly, implicitly, relatively, or underspecified). This information is then used to construct a timeline for each document (Alonso, 2008).

In the following section, we will redefine the concept of a temporal document profile and define geographic document profiles analogously. After explaining how two entries of the respective document profiles can be compared with each other in Section 4.4.2 and Section 4.4.3, respectively, we extend the concept for spatio-temporal events, i.e., introduce event document profiles in Section 4.4.4.

4.4.1 Temporal and Geographic Document Profiles

Analogously to document profiles in general, the idea behind so-called temporal and geographic document profiles is to describe and organize all the temporal and geographic information extracted from a document in a concise manner before such information is further utilized in search and exploration tasks. Since extracted temporal and geographic expressions have already been defined concisely in Section 4.3.3, we can formally define the respective document profiles as follows:

Definition 4.4. *(Temporal Document Profile)*
Given a document collection D, with each document $d_i \in D$ a *temporal document profile* $tdp(d_i)$ is associated containing all the extracted temporal expressions as defined in Definition 4.2.

Definition 4.5. *(Geographic Document Profile)*
Given a document collection D, with each document $d_i \in D$ a *geographic document profile* $gdp(d_i)$ is associated containing all the extracted geographic expressions as defined in Definition 4.3.

A temporal document profile is thus a set of extracted temporal expressions. Recall that an extracted temporal expression is a three-tuple $\langle t_i, s(t)_i, p(t)_i \rangle$, so that the entries can naturally be sorted by their occurrence positions in the document. However, the expressions can also be ordered based on their normalized semantics. While duration expressions can be sorted according to the length of the interval they are referring to, time and date expressions can be sorted according to when they occurred in time. For this, one has to compare the temporal expressions occurring in a document with each other independent of their granularities as will be described in Section 4.4.2.

Similarly, a geographic document profile is thus a set of extracted geographic expressions, i.e., a set of three-tuples of the form $\langle g_i, s(g)_i, p(g)_i \rangle$. Again, the natural way to sort the entries is based on the occurrence positions in the document, but they can also be organized based on the location they are referring to, and thus based on their normalized semantics. As for the entries of temporal document profiles, it is important that the entries of geographic document profiles can be compared to each other independent of the granularities of the single expressions. In Section 4.4.3, we will present a procedure to compare geographic expressions with each other.

4.4.2 Comparing Temporal Expressions

To analyze the content of temporal document profiles, there is a need to compare temporal expressions of the types date or time with each other. For this, one has to take care of the different granularities of temporal expressions as was exemplarily shown in Figure 2.1(c) (page 14) where the hierarchical organization of temporal expressions was explained. Independent of the granularity, the normalized value – in the following called *chronon* – of each temporal expression of the types date and time can be anchored in a timeline.

Timelines

For the representation of different granularities, we assume different timelines for each granularity, e.g., T_{day} for days and T_{month} for months. For example, "March 11, 2013" can be anchored in T_{day} and "September 2013" can be anchored in T_{month}. To explain how to compare two chronons with each other, we assume the following timelines $\mathcal{T} = \{T_{day}, T_{month}, T_{year}\}$ for day, month, and year, respectively. Of course, many more timelines exist (e.g., minute, hour, season, etc.) but – for the sake of simplicity – we use only these three timelines.

Temporal Precedence and Containment Relationships

To compare two chronons of the same granularity with each other, we introduce a *temporal precedence relationship*. Formally, this precedence relationship is defined as follows:

Definition 4.6. *(Temporal Precedence Relationship $\prec_T$)*
Using the *temporal precedence relationship* $\prec_T$, the relationship between two chronons $t_i \in \mathcal{T}'$, $t_j \in \mathcal{T}''$, with $\mathcal{T}' = \mathcal{T}''$, $t_i \neq t_j$, can be determined so that either $t_i \prec_T t_j$ or $t_j \prec_T t_i$.

All chronons of the same granularity, i.e., of the same timeline, can now be compared with each other. However, a document typically contains temporal expressions of different granularities so that there is a need to compare two chronons anchored in different timelines. For this, we introduce an additional relationship called *temporal containment relationship* that is formally defined as follows:

Definition 4.7. *(Temporal Containment Relationship $\subset_T$)*
Given two chronons $t_i \in \mathcal{T}'$ and $t_j \in \mathcal{T}''$, with $\mathcal{T}'$ being more fine grained than $\mathcal{T}''$. The *temporal containment relationship* $\subset_T$ between t_i and t_j holds ($t_i \subset_T t_j$) if t_i is contained in t_j.

Temporal Mapping Function

Now, chronons of the same granularity can be compared to each other, and chronons of different timelines can be checked for a containment relationship. However, two chronons can also be of different timelines without a containment relationship. To compare such chronons, we exploit the characteristic of temporal information that it can be organized hierarchically (cf. Section 2.3.1) and introduce a so-called *temporal mapping function*. This function can be applied to two chronons until they are of the same timeline so that the temporal precedence relationship can be checked for the mapped chronons. The temporal mapping function is defined as follows:

Definition 4.8. *(Temporal Mapping Function α_T)*
The *temporal mapping function* $\alpha_T(t_i') = t_i''$ maps the chronon $t_i' \in T'$ to the next coarser timeline, so that $t_i'' \in T''$, with T'' being the next coarser timeline of T' in the temporal hierarchy.

Algorithm 4.1 Procedure to compare two chronons of any timelines using the temporal precedence relationship $\prec_T$, the temporal containment relationship $\subset_T$, and the temporal mapping function α_T.

```
1:  procedure COMPARE_CHRONONS(t_1, t_2)
2:      t_1^* = t_1, t_2^* = t_2                          ▷ keep original values of t_1 and t_2
3:      if (t_1.timeline < t_2.timeline) and (t_1 ⊂_T t_2) then
4:          return t_1^* contained in t_2^*
5:      else if (t_2.timeline < t_1.timeline) and (t_2 ⊂_T t_1) then
6:          return t_2^* contained in t_1^*
7:      end if
8:      while (t_1.timeline < t_2.timeline) do
9:          t_1 = α_T(t_1)
10:     end while
11:     while (t_2.timeline < t_1.timeline) do
12:         t_2 = α_T(t_2)
13:     end while
14:     if t_1 ≺_T t_2 then
15:         return t_1^* before t_2^*
16:     else if t_2 ≺_T t_1 then
17:         return t_1^* after t_2^*
18:     else
19:         return t_1^* equals t_2^*
20:     end if
21: end procedure
```

Assuming the three example timelines $\mathcal{T} = \{T_{day}, T_{month}, T_{year}\}$, chronons of the granularity day can be mapped to the month timeline, and chronons of the granularity month can be mapped to the year timeline by applying the temporal mapping function. For instance, $\alpha_T(\text{“2013-03-11”}) = \text{“2013-03”}$ and $\alpha_T(\text{“2013-09”}) = \text{“2013”}$. Of course, the temporal mapping function can also be applied recursively, e.g., $\alpha_T(\alpha_T(\text{“2013-03-11”})) = \text{“2013”}$.

Algorithm to Compare Chronons

Combining the temporal precedence relationship (Definition 4.6), the temporal containment relationship (Definition 4.7), and the temporal mapping function (Definition 4.8), all chronons can be compared with each other independent of their granularities by applying the procedure described in Algorithm 4.1.

In lines 3 to 7, the two chronons are checked for a containment relationship. If there is no containment relationship, t_1 and t_2 are mapped to the same timeline in lines 8 to 13.[7] Then, in lines 14 to 20, the relationship between the two chronons is determined, and the algorithm returns this relationship.

[7] For the sake of simplicity, we assume that the temporal hierarchy is linear, i.e., the timelines are organized linearly as in the case of our three example timelines of $\mathcal{T} = \{T_{day}, T_{month}, T_{year}\}$. However, if the hierarchies were more complex (parallel), the algorithm would only have to be slightly modified: (i) the containment relationship check (lines 3 to 7) would be applied to linearly related chronons only; (ii) instead of checking the timelines of t_1 and t_2 to be identical (8 to 13), one would map two non-linear related chronons up to their common governor timeline resulting in “(close to) overlap” relationships, which were to be distinguished from the equal relationship. In addition, we assume that any temporal hierarchy has a single common root timeline even if the hierarchy is not organized completely linear.

t_1	t_2	timelines	mappings	relation
2013	2013	T_{year}, T_{year}	–	$t_1 = t_2$
2013	2014	T_{year}, T_{year}	–	$t_1 \prec_T t_2$
2013	2013-09	T_{year}, T_{month}	$\alpha_T(t_2)$	$t_2 \subset_T t_1$
2014	2013-09	T_{year}, T_{month}	$\alpha_T(t_2)$	$t_2 \prec_T t_1$
2013	2013-03-11	T_{year}, T_{day}	$\alpha_T(\alpha_T(t_2))$	$t_2 \subset_T t_1$
2013	2013-09-13	T_{year}, T_{day}	$\alpha_T(\alpha_T(t_2))$	$t_2 \subset_T t_1$
2014	2013-03-11	T_{year}, T_{day}	$\alpha_T(\alpha_T(t_2))$	$t_2 \prec_T t_1$
2014	2013-09-13	T_{year}, T_{day}	$\alpha_T(\alpha_T(t_2))$	$t_2 \prec_T t_1$
2013-09	2013-03-11	T_{month}, T_{day}	$\alpha_T(t_2)$	$t_2 \prec_T t_1$
2013-09	2013-09-13	T_{month}, T_{day}	$\alpha_T(t_2)$	$t_2 \subset_T t_1$
2013-03-11	2013-09-13	T_{day}, T_{day}	$\alpha_T(t_1), \alpha_T(t_2)$	$t_1 \prec_T t_2$

Table 4.2: Examples showing how to compare two chronons with each other to determine their temporal relationship. If both chronons are identical, there is an equal relationship between t_1 and t_2 without any mappings as exemplarily shown for $t_1 = t_2 = 2013$.

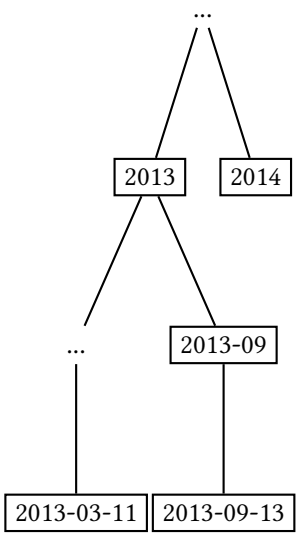


Figure 4.2: Hierarchy structure of the example chronons.

Note that our algorithm to compare chronons does not check for all thirteen temporal relations defined by Allen (1983) as briefly introduced in Section 2.3.1. Obviously, the relationship can be determined as equal (line 19) and as before and its inverse (lines 15 and 17). However, we do not distinguish between Allen's "before" and "meet" relations. In addition, the containment relations (lines 4 and 6) cover Allen's relations "during", "starts", "finishes" and their inverses. Finally, Allen's overlap relations are not considered here since they can only occur if the temporal hierarchy is not linear or if intervals are compared to chronons. While this will be relevant in Chapter 5, when discussing information retrieval with temporal constraints (i.e., query intervals), comparing intervals to chronons is not relevant for comparing date and time expressions in a temporal document profile.

In Table 4.2, we present some examples how two chronons of our three example granularities are compared with each other and what relationships hold between them. For these examples, we use the following five chronons "2013", "2014", "2013-09", "2013-03-11", and "2013-09-13". In Figure 4.2, their temporal hierarchy structure is depicted. By applying Algorithm 4.1 to each pair of chronons, we can determine the temporal relations between all chronons.

Mapping Chronons for Equality

A further procedure that becomes important in Chapter 6 is the mapping of two chronons for equality. While the comparison of two chronons described in Algorithm 4.1 allows to order chronons, this procedure determines how similar two chronons of any timeline are, based on their hierarchical distance. The main assumption is that the smaller the timelines of two chronons, the more similar they can be. As for Algorithm 4.1, the hierarchical organization of chronons is again exploited and the temporal mapping function as well as the temporal precedence relationship are applied. The *mapping chronons for equality* procedure is described in Algorithm 4.2.

Algorithm 4.2 Procedure to map two chronons of any timelines until they are equal. The procedure makes use of the temporal precedence relationship $\prec_T$ and the temporal mapping function α_T.

```
1:  procedure MAP_CHRONONS_FOR_EQUALITY(t_1, t_2)
2:      map_1 = 0, map_2 = 0                          ▷ tracking the mapping steps of t_1 and t_2
3:      t_1^* = t_1, t_2^* = t_2                      ▷ keep original values of t_1 and t_2
4:      while (t_1.timeline < t_2.timeline) do
5:          t_1 = α_T(t_1)
6:          map_1 = map_1 + 1
7:      end while
8:      while (t_2.timeline < t_1.timeline) do
9:          t_2 = α_T(t_2)
10:         map_2 = map_2 + 1
11:     end while
12:     while (t_1 ≺_T t_2) or (t_2 ≺_T t_1) do
13:         t_1 = α_T(t_1)
14:         t_2 = α_T(t_2)
15:         map_1 = map_1 + 1
16:         map_2 = map_2 + 1
17:     end while
18:     return t_1^* equals t_2^* after map_1 and map_2 mappings on timeline t_1.timeline
19: end procedure
```

Similar to the procedure to compare two chronons, this procedure maps the chronons t_1 and t_2 to the same timeline (lines 4 to 11). Then, however, the chronons are further mapped until they are equal, i.e., neither $t_1 \prec_T t_2$ nor $t_2 \prec_T t_1$ (lines 12 to 17). In general, two chronons are more similar to each other the less mappings are necessary and the finer the timeline of the chronons once they are equal. Thus, it is important to track the number of required mapping steps (lines 2, 6, 10, 15, and 16).

In Table 4.3, we show some examples how to map two chronons for equality. For this, we again use the five chronons "2013", "2014", "2013-09", "2013-03-11", and "2013-09-13" for which the hierarchy structure is depicted in Figure 4.2. In Table 4.3, it is listed how many mapping steps are necessary to make the two chronons being equal and on which timeline the equal relationship holds. Note that in addition to the three timelines for day, month, and year, we assume a forth timeline T_{global} on which all chronons become equal. This timeline can be considered as the root of the temporal hierarchy. Thus, the chronons can match on one of the following four timelines $\mathcal{T} = \{T_{day}, T_{month}, T_{year}, T_{global}\}$. Obviously, if the equal relationship between two chronons is determined in T_{global}, the two chronons are not very similar compared to chronons that have been matched earlier.

The examples in Table 4.3 show, for instance, that "2013-09" and "2013-09-13" are quite similar since only one mapping is necessary and they are already equal in T_{month}. In contrast, although there is only one mapping required for "2013" and "2013-09", they can be considered less similar since they are equal not till T_{year}. Finally, "2014" and "2013-03-11" are not very similar at all since a total of four mappings is necessary to make them equal in T_{global}.

t_1	t_2	timelines	mappings	equal at	map_1	map_2
2013	2013	T_{year}, T_{year}	–	T_{year}	0	0
2013	2014	T_{year}, T_{year}	$\alpha_T(t_1), \alpha_T(t_2)$	T_{global}	1	1
2013	2013-09	T_{year}, T_{month}	$\alpha_T(t_2)$	T_{year}	0	1
2014	2013-09	T_{year}, T_{month}	$\alpha_T(t_1), \alpha_T(\alpha_T(t_2))$	T_{global}	1	2
2013	2013-03-11	T_{year}, T_{day}	$\alpha_T(\alpha_T(t_2))$	T_{year}	0	2
2013	2013-09-13	T_{year}, T_{day}	$\alpha_T(\alpha_T(t_2))$	T_{year}	0	2
2014	2013-03-11	T_{year}, T_{day}	$\alpha_T(t_1), \alpha_T(\alpha_T(\alpha_T(t_2)))$	T_{global}	1	3
2014	2013-09-13	T_{year}, T_{day}	$\alpha_T(t_1), \alpha_T(\alpha_T(\alpha_T(t_2)))$	T_{global}	1	3
2013-09	2013-03-11	T_{month}, T_{day}	$\alpha_T(t_1), \alpha_T(\alpha_T(t_2))$	T_{year}	1	2
2013-09	2013-09-13	T_{month}, T_{day}	$\alpha_T(t_2)$	T_{month}	0	1
2013-03-11	2013-09-13	T_{day}, T_{day}	$\alpha_T(\alpha_T(t_1)), \alpha_T(\alpha_T(t_2))$	T_{year}	2	2

Table 4.3: Examples showing how to map two chronons for equality. If both chronons are identical, there is an equal relationship between t_1 and t_2 without any mappings on the original timeline as exemplarily shown for t_1 = t_2 = 2013.

Summary

In this section, we introduced two algorithms to compare two chronons with each other. While the first algorithm determines the temporal relationship between two chronons, the second algorithm determines how similar two chronons are based on their distance in the temporal hierarchy. Both algorithms will become important when comparing spatio-temporal events with each other (Section 4.4.5), but also when describing spatio-temporal and event-centric search and exploration tasks in Chapter 5 and Chapter 6.

4.4.3 Comparing Geographic Expressions

Many geo-taggers that extract geographic expressions from text documents only assign point geometry information to a location as the only geometry information, regardless of the actual geographic extent of the location. However, as already mentioned in Section 4.3.3, containment information about locations is also often associated with the extracted locations. For this, we exploit this containment and thus the granularity information of the locations to compare two geographic expressions.

Similar to the structure of the previous section, in the following, we will present geographic relationships and the geographic mapping function as well as two algorithms to compare geographic expressions with each other.

Geographic Granularities

As was explained in Section 2.4.1 and exemplarily shown in Figure 2.2(c) (page 18), geographic expressions can be organized hierarchically similar to temporal expressions. Thus, every geographic expression can be associated with a specific granularity such as city or country. We assume the following geographic granularities $\mathcal{G} = \{G_{city}, G_{state}, G_{country}\}$, e.g., “Leipzig” can be anchored in G_{city}, “Saxony” can be

anchored in G_{state}, and "Germany" can be anchored in $G_{country}$. Although many more geographic granularities exist (e.g., address, suburb, county, etc.), we assume – for the sake of simplicity – only these three geographic granularities to explain how we compare two locations.

Geographic Disconnect and Containment Relationships

To compare two locations of the same granularity with each other, we introduce a *geographic disconnect relationship*. Formally, this disconnect relationship is defined as follows:

Definition 4.9. *(Geographic Disconnect Relationship $\emptyset_G$)*
Using the *geographic disconnect relationship* $\emptyset_G$, the relationship between two locations $g_i \in \mathcal{G}'$, $g_j \in \mathcal{G}''$, with $\mathcal{G}' = \mathcal{G}''$, can be determined as $g_i \emptyset_G g_j$, if $g_i \neq g_j$.

Due to the hierarchical organization of geographic information, two locations of the same granularity are either equal or geographically disconnected. However, instead of comparing only locations of the same granularity with each other, it is necessary to also compare locations of different granularities since typically documents and thus geographic document profiles contain locations of several granularities. For that, we introduce a *geographic containment relationship* that is formally defined as follows:

Definition 4.10. *(Geographic Containment Relationship $\subset_G$)*
Given two locations $g_i \in \mathcal{G}'$ and $g_j \in \mathcal{G}''$, with $\mathcal{G}'$ being more fine grained than $\mathcal{G}''$. The *geographic containment relationship* $\subset_G$ between g_i and g_j holds ($g_i \subset_G g_j$) if g_i is contained in g_j.

Note that one could also determine whether or not a containment relationship holds between two locations based on explicit region information (specified, e.g., in the form of a polygon). However, as already mentioned above, the hierarchical containment information is typically accessible using the gazetteer of a geo-tagger while explicit polygonal information is often not available. Thus, we rely on the containment information rather than explicit polygonal information about the locations.

Geographic Mapping Function

Now, locations of the same granularity can be compared to each other, and locations of different granularities can be checked for a containment relationship. However, two locations can be of different granularities without a containment relationship. Although in the case of a linear geographic hierarchy as in our example ($G = \{G_{city}, G_{state}, G_{country}\}$) there would either be a containment or a disconnect relationship between any two locations, in the case of a non-linear hierarchy, there could also be partially overlapping locations. Thus, and since it is required for the "mapping to equality" procedure for locations that will be described below, we introduce a geographic mapping function that is defined as follows:

Definition 4.11. *(Geographic Mapping Function α_G)*
The *geographic mapping function* $\alpha_G(g_i') = g_i''$ maps the location $g_i' \in G'$ to the next coarser geographic granularity, so that $g_i'' \in G''$, with G'' being the next coarser granularity of G' in the geographic hierarchy.

Assuming the three example geographic granularities $\mathcal{G} = \{G_{city}, G_{state}, G_{country}\}$, locations of the granularities city and state can be mapped to the state and country granularities, respectively, by applying the geographic mapping function. For example, α_G("Leipzig") = "Saxony" and α_G("Saxony") =

Algorithm 4.3 Procedure to compare two locations of any granularities making use of the geographic disconnect $\emptyset_G$ and containment $\subset_G$ relationships, and the geographic mapping function α_G.

```
1:  procedure COMPARE_LOCATIONS(g1, g2)
2:      g1* = g1, g2* = g2                          ▷ keep original values of g1 and g2
3:      if (g1.granularity < g2.granularity) and (g1 ⊂G g2) then
4:          return g1* contained in g2*
5:      else if (g2.granularity < g1.granularity) and (g2 ⊂G g1) then
6:          return g2* contained in g1*
7:      end if
8:      while (g1.granularity < g2.granularity) do
9:          g1 = αG(g1)
10:     end while
11:     while (g2.granularity < g1.granularity) do
12:         g2 = αG(g2)
13:     end while
14:     if (g1 ∅G g2) then
15:         return g1* disconnected of g2*
16:     else
17:         return g1* equals g2*
18:     end if
19: end procedure
```

"Germany". Of course, as the temporal mapping function, the geographic mapping function can also be applied recursively, e.g., $\alpha_G(\alpha_G(\text{"Leipzig"})) =$ "Germany". Since the same example locations will be used below when explaining the algorithms for comparing locations and determining their similarity, the hierarchy structure of the locations is depicted in Figure 4.3.

Algorithm to Compare Locations

Similar to Algorithm 4.1 to compare chronons with each other, we describe in Algorithm 4.3 the procedure to compare two locations with each other independent of their granularities. In this procedure, the geographic disconnect relationship (Definition 4.9) and the geographic containment relationship (Definition 4.10) as well as the geographic mapping function (Definition 4.11) will be used.

In lines 3 to 7, the two locations g_1 and g_2 are checked for a containment relationship. If there is no containment relationship, both locations are mapped to the same granularity in lines 8 to 13.[8] Then, in lines 14 to 18, the geographic relationship between g_1 and g_2 is determined as either equal of disconnected.

In contrast to chronons, which can be chronologically ordered, locations can only be distinguished to be either equal, disconnected or contained in each other. In addition, note that not all the geographic

[8] As for the corresponding algorithm for chronons, for the sake of simplicity, we assume that the geographic hierarchy is linear. Thus, the algorithm would have to be slightly modified if the hierarchy was more complex: (i) the containment relationship (lines 3 to 7) would be applied to linearly related locations only; (ii) instead of checking for the granularities of g_1 and g_2 to be identical (8 to 13), one would map two non-linear related locations up to their common governor granularity resulting in "(close to) overlap" relationships, which were to be distinguished from the equal relationship. Again, we assume that any geographic hierarchy has a single common root granularity even if the hierarchy is not organized completely linear.

g_1	g_2	granularities	mappings	relation
Germany	Germany	$G_{country}, G_{country}$	–	$g_1 = g_2$
Germany	Spain	$G_{country}, G_{country}$	–	$g_1 \emptyset_G g_2$
Germany	Saxony	$G_{country}, G_{state}$	–	$g_2 \subset_G g_1$
Spain	Saxony	$G_{country}, G_{state}$	$\alpha_G(g_2)$	$g_1 \emptyset_G g_2$
Germany	Heidelberg	$G_{country}, G_{city}$	–	$g_2 \subset_G g_1$
Germany	Leipzig	$G_{country}, G_{city}$	–	$g_2 \subset_G g_1$
Spain	Heidelberg	$G_{country}, G_{city}$	$\alpha_G(\alpha_G(g_2))$	$g_1 \emptyset_G g_2$
Spain	Leipzig	$G_{country}, G_{city}$	$\alpha_G(\alpha_G(g_2))$	$g_1 \emptyset_G g_2$
Saxony	Heidelberg	G_{state}, G_{city}	$\alpha_G(g_2)$	$g_1 \emptyset_G g_1$
Saxony	Leipzig	G_{state}, G_{city}	–	$g_2 \subset_G g_1$
Heidelberg	Leipzig	G_{city}, G_{city}	–	$g_1 \emptyset_G g_2$

Table 4.4: Examples showing how to compare two locations with each other to determine their geographic relationship. If both locations are identical, there is an equal relationship between g_1 and g_2 without mappings as exemplarily shown for $g_1 = g_2$ = Germany.

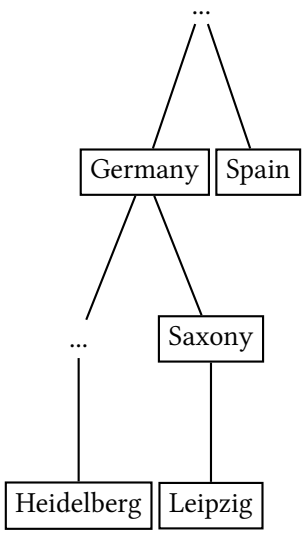


Figure 4.3: Hierarchy structure of the example locations.

relations described in Section 2.4.1 are considered in Algorithm 4.3. The four relations of the region connection calculus 8 (RCC8) "tangential proper part", "non-tangential proper part" and their inverses are all captured as containment relationship. Furthermore, while the algorithm's "equal" relationship is identical to the RCC8 "equal" relationship, the RCC8 relations "disconnected" and "externally connected" are both considered as disconnected by the algorithm. Finally, the "partially overlapped" relationship is not considered because it can only occur if the geographic hierarchy is not linear or if arbitrary regions are compared to locations of specified granularities. As the temporal overlap relation, the geographic overlap relation will also become relevant in Chapter 5 when information retrieval with temporal and geographic constraints are developed.

In Table 4.4, we show some examples how two locations can be compared by applying Algorithm 4.3. In Figure 4.3, the geographic hierarchy of the following five example locations is depicted: "Germany" $\in G_{country}$, "Spain" $\in G_{country}$, "Saxony" $\in G_{state}$, "Heidelberg" $\in G_{city}$, and "Leipzig" $\in G_{city}$. Comparing any two of the five locations with each other, there is either an "equal", disconnected ($\emptyset_G$), or containment ($\subset_G$) relationship.

Mapping Locations for Equality

In Algorithm 4.4, the *mapping locations for equality* procedure is shown. As its temporal counterpart (Algorithm 4.2), it will become important in Chapter 6. Its structure is also quite similar and the goal is analogous to determine the similarity between two locations of any granularities. To decide how similar two locations are based on their hierarchical distance, the number of necessary mapping steps and the granularity when the two locations match are determined. The less mappings are necessary and the finer the granularity when the two locations match, the more similar are the two locations.

Algorithm 4.4 Procedure to map two locations of any granularities until they are equal. The procedure makes use of the geographic disconnect relationship $\emptyset_G$ and the geographic mapping function α_G.

procedure MAP_LOCATIONS_FOR_EQUALITY(t_1, t_2)
  $map_1 = 0, map_2 = 0$ ▷ tracking the mapping steps of g_1 and g_2
  $g_1^* = g_1, g_2^* = g_2$ ▷ keep original values of g_1 and g_2
  while ($g_1.granularity < g_2.granularity$) **do**
    $g_1 = \alpha_G(g_1)$
    $map_1 = map_1 + 1$
  end while
  while ($g_2.granularity < g_1.granularity$) **do**
    $g_2 = \alpha_G(g_2)$
    $map_2 = map_2 + 1$
  end while
  while ($g_1 \emptyset_G g_2$) **do**
    $g_1 = \alpha_G(g_1)$
    $g_2 = \alpha_G(g_2)$
    $map_1 = map_1 + 1$
    $map_2 = map_2 + 1$
  end while
  return g_1^* equals g_2^* after map_1 and map_2 mappings on granularity $g_1.granularity$
end procedure

In Table 4.5, we show how our five example locations are compared, assuming the geographic granularities $\mathcal{G} = \{G_{city}, G_{state}, G_{country}, G_{global}\}$, with G_{global} being the root granularity of the geographic hierarchy on which all locations become equal.

Summary

In this section, we introduced two algorithms to compare two geographic locations with each other. While the first algorithm determines the geographic relationship between two locations, the second algorithm determines how similar two locations are. Both algorithms will become important when describing spatio-temporal and event-centric search and exploration tasks in Chapter 5 and Chapter 6, but also when comparing spatio-temporal events with each other (Section 4.4.5). However, before we explain how to compare spatio-temporal events with each other, we first explain how spatio-temporal events can be organized by introducing the concept of event document profiles.

4.4.4 Event Document Profile

While the temporal and geographic document profiles defined above (Definition 4.4 and Definition 4.5, respectively) will be valuable for the spatio-temporal search and exploration scenarios, we now introduce event document profiles of which we will make use for several event-centric search and exploration scenarios. Formally, an event document profile is defined as follows:

Definition 4.12. *(Event Document Profile)*
Given a document collection D, each document $d_i \in D$ is associated with an *event document profile* $edp(d_i)$ containing all the extracted spatio-temporal events as defined in Definition 4.1.

g_1	g_2	granularities	mappings	equal at	map_1	map_2
Germany	Germany	$G_{country}, G_{country}$	–	$G_{country}$	0	0
Germany	Spain	$G_{country}, G_{country}$	$\alpha_G(g_1), \alpha_G(g_2)$	G_{global}	1	1
Germany	Saxony	$G_{country}, G_{state}$	$\alpha_G(g_2)$	$G_{country}$	0	1
Spain	Saxony	$G_{country}, G_{state}$	$\alpha_G(g_1), \alpha_G(\alpha_G(g_2))$	G_{global}	1	2
Germany	Heidelberg	$G_{country}, G_{city}$	$\alpha_G(\alpha_G(g_2))$	$G_{country}$	0	2
Germany	Leipzig	$G_{country}, G_{city}$	$\alpha_G(\alpha_G(g_2))$	$G_{country}$	0	2
Spain	Heidelberg	$G_{country}, G_{city}$	$\alpha_G(g_1), \alpha_G(\alpha_G(\alpha_G(g_2)))$	G_{global}	1	3
Spain	Leipzig	$G_{country}, G_{city}$	$\alpha_G(g_1), \alpha_G(\alpha_G(\alpha_G(g_2)))$	G_{global}	1	3
Saxony	Heidelberg	G_{state}, G_{city}	$\alpha_G(g_1), \alpha_G(\alpha_G(g_2))$	$G_{country}$	1	2
Saxony	Leipzig	G_{state}, G_{city}	$\alpha_G(g_2)$	G_{city}	0	1
Heidelberg	Leipzig	G_{city}, G_{city}	$\alpha_G(\alpha_G(g_1)), \alpha_G(\alpha_G(g_2))$	$G_{country}$	2	2

Table 4.5: Examples showing how to map two locations for equality. If both locations are identical, there is an equal relationship between g_1 and g_2 without any mappings on the original granularity as shown for g_1 = g_2 = Germany.

An event document profile is thus defined analogously to temporal document profiles and geographic document profiles (cf. Section 4.4.1) and contains all the tuples of extracted temporal and geographic expressions, which are determined as forming spatio-temporal events. While the content of event document profiles thus depends on the applied method to extract spatio-temporal events, all events can naturally be ordered by their occurrence position in the document. Of course, the events can also be organized based on their temporal or geographic semantics.

4.4.5 Comparing Spatio-temporal Events

Using the above described algorithms, and thus the temporal and geographic relationships and mapping functions defined in Section 4.4.2 and Section 4.4.3, two spatio-temporal events can be temporally and geographically compared with each other. In addition, two events can also be mapped for equality.

Temporal and Geographic Relationships between Events

Assuming two spatio-temporal events $e_1 = \langle te_1, ge_1 \rangle$ and $e_2 = \langle te_2, ge_2 \rangle$, then the temporal relationship between e_1 and e_2 is determined by applying the Compare_Chronons procedure (Algorithm 4.1) to the chronons t_1 and t_2 of the temporal components te_1 and te_2 resulting in one of the following relationships: either the events are temporally identical ($t_1 = t_2$), one event is temporally contained in the other event ($t_1 \subset_T t_2$ or $t_2 \subset_T t_1$), or one event temporally precedes the other event ($t_1 \prec_T t_2$ or $t_2 \prec_T t_1$).

Similarly as for the temporal relationship, the geographic relationship between $e_1 = \langle te_1, ge_1 \rangle$ and $e_2 = \langle te_2, ge_2 \rangle$ is determined by applying the Compare_Locations procedure (Algorithm 4.3) to the locations g_1 and g_2 of the geographic components ge_1 and ge_2 resulting in one of the following relationships: either the two events are geographically equal ($g_1 = g_2$), or disconnected ($g_1 \emptyset_G g_2$), or one event is geographically contained in the other event ($g_1 \subset_G g_2$ or $g_2 \subset_G g_1$).

Mapping Events for Equality

To determine how similar two spatio-temporal events are based on their temporal and geographic hierarchical distances, we combine the two procedures Map_Chronons_For_Equality (Algorithm 4.2) and Map_Locations_For_Equality (Algorithm 4.4). As for determining the similarity of temporal expressions and geographic expressions, we assume that two events $e_1 = \langle te_1, ge_1 \rangle$ are the more similar, the less temporal and geographic mappings are necessary and the finer the timelines and geographic granularities when the temporal and geographic components of e_1 and e_2 match.

In Table 4.6, we present some examples to compare spatio-temporal events with each other. For this, we again assume the timelines $\mathcal{T} = \{T_{day}, T_{month}, T_{year}, T_{global}\}$ and the geographic granularities $\mathcal{G} = \{G_{city}, G_{state}, G_{country}, G_{global}\}$. Furthermore, we use seven spatio-temporal events with the temporal and geographic components of the previous examples so that the temporal and geographic hierarchies depicted in Figure 4.2 (page 140) and Figure 4.3 (page 145), respectively, can be used to follow the mappings in our examples.

While some event pairs have to be mapped to T_{global} and G_{global} and are thus not very similar (e.g., all event pairs with one event being ⟨"2014", "Spain"⟩), other event pairs match on finer timelines and granularities. For instance, ⟨"2013-03-11", "Leipzig"⟩ and ⟨"2013-03-11", "Heidelberg"⟩ are temporally equal on T_{day} and match geographically on $G_{country}$. Similarly, when comparing $e_1 = \langle$"2013-09", "Germany"$\rangle$ with $e_2 = \langle$"2013-09-13", "Saxony"$\rangle$, e_2 is temporally and geographically contained in e_1. Since only the components of e_2 have to be mapped and the matching timelines and granularities are T_{month} and $G_{country}$, the two events are relatively similar to each other in contrast to several other event pairs shown in Table 4.6.

Obviously, "relatively similar" is a rather less concise description of the similarity between events. In Chapter 6, we will develop a concise similarity model for events, which relies on the mapping to equality algorithms described above.

4.5 Event Extraction

After having motivated and defined our concept of spatio-temporal events, and after having explained temporal, geographic, and event document profiles as well as different methods to compare their entries with each other, this section treats another important aspect of spatio-temporal events, namely their extraction from textual documents.

Recall that a spatio-temporal event consists of an extracted temporal and an extracted geographic expression, which have to cooccur within a specified window in a document (e.g., within a sentence), and, which have to be used in the text to refer to the location and the point in time where and when something is, was, or will be happening (cf. Definition 4.1). In addition, in Section 4.3, we discussed examples of potential spatio-temporal events and explained why some combinations of temporal and geographic expressions form valid spatio-temporal events while other combinations do not result in valid events.

In the following, we assume that the temporal expression and the geographic expression have to cooccur within a sentence. Then, the simplest approach to extract spatio-temporal events is to rely on cooccurrences on sentence level. This recall-optimized approach is detailed in Section 4.5.1 where we also present a description of the processing pipeline for spatio-temporal event extraction.

event e_1		event e_2						
t_1	g_1	t_2	g_2	equal at	map_{t1}	map_{g1}	map_{t2}	map_{g1}
2013	Germany	2014	Spain	T_{global}, G_{global}	1	1	1	1
2013	Germany	2013	Heidelberg	$T_{year}, G_{country}$	0	0	0	2
2013	Germany	2013-09	Germany	$T_{year}, G_{country}$	0	0	1	0
2013	Germany	2013-03-11	Leipzig	$T_{year}, G_{country}$	0	0	2	2
2013	Germany	2013-03-11	Heidelberg	$T_{year}, G_{country}$	0	0	2	2
2013	Germany	2013-09-13	Saxony	$T_{year}, G_{country}$	0	0	2	1
2014	Spain	2013	Heidelberg	T_{global}, G_{global}	1	1	1	3
2014	Spain	2013-09	Germany	T_{global}, G_{global}	1	1	2	1
2014	Spain	2013-03-11	Leipzig	T_{global}, G_{global}	1	1	3	3
2014	Spain	2013-03-11	Heidelberg	T_{global}, G_{global}	1	1	3	3
2014	Spain	2013-09-13	Saxony	T_{global}, G_{global}	1	1	3	2
2013	Heidelberg	2013-09	Germany	$T_{year}, G_{country}$	0	2	1	0
2013	Heidelberg	2013-03-11	Leipzig	$T_{year}, G_{country}$	0	2	2	2
2013	Heidelberg	2013-03-11	Heidelberg	T_{year}, G_{city}	0	0	2	0
2013	Heidelberg	2013-09-13	Saxony	$T_{year}, G_{country}$	0	2	2	1
2013-09	Germany	2013-03-11	Leipzig	$T_{year}, G_{country}$	1	0	2	2
2013-09	Germany	2013-03-11	Heidelberg	$T_{year}, G_{country}$	1	0	2	2
2013-09	Germany	2013-09-13	Saxony	$T_{month}, G_{country}$	0	0	1	1
2013-03-11	Leipzig	2013-03-11	Heidelberg	$T_{day}, G_{country}$	0	2	0	2
2013-03-11	Leipzig	2013-09-13	Saxony	T_{year}, G_{state}	2	1	2	0
2013-03-11	Heidelberg	2013-09-13	Saxony	$T_{year}, G_{country}$	2	2	2	2

Table 4.6: Examples how to map two spatio-temporal events for equality. If both events are identical, there is a geographic equal relationship between g_1 and g_2 and a temporal equal relationship between t_1 and t_2 without any mappings on the original timelines and geographic granularities.

To estimate the extraction quality of the cooccurrence approach, a data set containing manually annotated spatio-temporal events is required. Thus, after explaining our annotation guidelines for spatio-temporal events in Section 4.5.2, we present in Section 4.5.3 our newly created annotated data sets together with a description of the annotation procedure and the performance of the cooccurrence approach.

Note that not a single data set is created but an initial, large data set and several smaller ones. This is necessary because we also experiment with further event extraction methods to improve the extraction quality. While the initial data set is used for the development of the more sophisticated extraction approaches, the smaller data sets are used for their evaluation. In Section 4.5.4 and Section 4.5.5, we detail the development of heuristic and linguistically-motivated approaches, respectively. Finally, in Section 4.5.6, we evaluate and compare the different approaches and discuss their advantages and disadvantages.

4.5.1 Extracting Spatio-temporal Events as Cooccurrences

The task of extracting spatio-temporal events is composed of several subtasks. Obviously, temporal expressions and geographic expressions have to be extracted and normalized, and it has to be determined which expressions cooccur in the same sentence. Thus, sentence splitting (also sometimes referred to as sentence boundary detection) is a subtask. Furthermore, the temporal tagging and geo-tagging tasks typically require some linguistic preprocessing. For instance, as described in Chapter 3, before HeidelTime can be applied for temporal tagging, the preprocessing steps of sentence splitting, tokenization, and part-of-speech tagging have to be performed. Finally, it should also be possible to process documents of different sources and to either output or store the extracted spatio-temporal events in any format depending on the requirements. Thus, several NLP tools and general tasks are involved in the event extraction process, and it is intuitive to organize the extraction following the pipeline principle.

As described in Section 2.5, UIMA is a framework and management architecture for deploying text processing pipelines and allows to easily combine different tools which have originally not been built to be used together. For the extraction of spatio-temporal events from text documents, we thus also make use of UIMA and explain in the following the components of our UIMA pipeline for the extraction of spatio-temporal events. Although we present this pipeline for the cooccurrence approach only, it is straightforward how the pipeline has to be extended for the more complex event extraction approaches.

Since spatio-temporal events share the key characteristics of temporal and geographic expressions, they can be normalized and are thus language-independent. To extract events from documents of different languages, a temporal tagger and a geo-tagger for the respective languages have to be applied. Thus, we apply our temporal tagger HeidelTime together with the required linguistic preprocessing tool for sentence splitting, tokenization, and part-of-speech tagging. As described in Section 3.5.8, the UIMA HeidelTime kit contains wrappers for preprocessing tools for all languages supported by HeidelTime.

Similar as for temporal taggers, there is no large variety of publicly available, multilingual geo-taggers (cf. Section 2.4.3). An available tool for multilingual geo-tagging, which also provides a lot of information about extracted locations – such as hierarchical containment information, which is crucial for comparing two spatio-temporal events with each other – is Yahoo! Placemaker.[9] Thus, we wrote a UIMA wrapper that calls the Yahoo! Placemaker Web service, annotates all extracted locations, and assigns normalization information to each geographic expression.

In Figure 4.4, the UIMA processing pipeline for event extraction with the cooccurrence approach is depicted. In addition to a document- or corpus-specific collection reader, and a CAS consumer for the final processing, four analysis engines are applied. The TreeTagger wrapper for linguistic preprocessing adds sentence, token, and part-of-speech annotations to the CAS objects, HeidelTime as temporal tagger adds TIMEX3 annotations, and Yahoo! Placemaker as geo-tagger adds extracted location information. Finally, the Cooccurrence Extractor iterates over each sentence and annotates each pair of temporal (of type "date" or "time") and geographic expressions cooccurring in a sentence as spatio-temporal event.

[9] Yahoo! Placemaker is officially not available anymore (`http://developer.yahoo.com/geo/placemaker/`). It was replaced by Yahoo! BOSS Geo Services containing the two main components Placefinder and PlaceSpotter, with the latter one performing the extraction and normalization of geographic expressions from textual documents in the same fashion as Yahoo! Placemaker. For further information, we refer to `https://developer.yahoo.com/boss/geo/` [last accessed October 1, 2014]. However, for the experiments described in this thesis, which rely on geographic expressions, we used the Yahoo! Placemaker Web service.

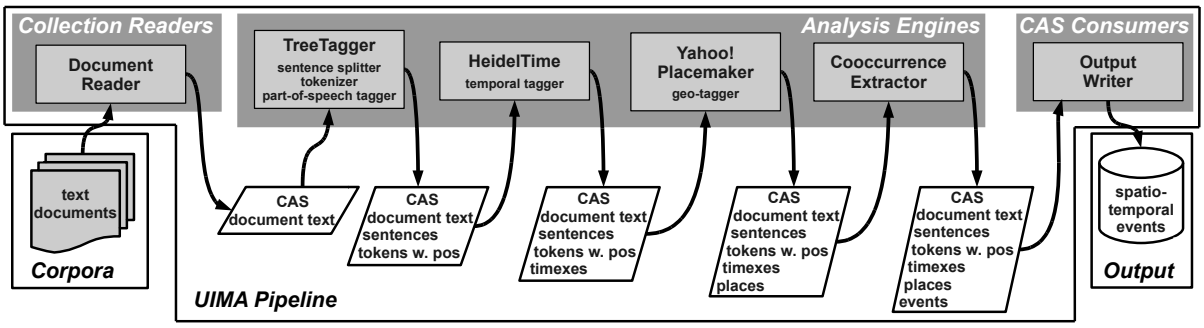


Figure 4.4: UIMA event extraction pipeline. After documents are read by a collection reader, four analysis engines are applied to finally extract spatio-temporal events with the cooccurrence approach. The CAS consumer performs the final processing, e.g., to store extracted events in a database.

Note that depending on the language, instead of the TreeTagger wrapper, another wrapper is used for linguistic preprocessing, e.g., the Stanford POS Tagger wrapper for Arabic. However, due to the pipeline principle and since different tools can be easily combined with each other, the TreeTagger wrapper can just be replaced, and all other components of the pipeline remain as shown in Figure 4.4.

To get an impression of the quality of spatio-temporal events extracted with the cooccurrence approach, we perform an evaluation on a corpus containing manually annotated spatio-temporal events. For the development of this corpus, we set up concise annotation guidelines as detailed in the following.

4.5.2 Guidelines for Annotating Spatio-temporal Events

Since we introduced our own concept of spatio-temporal events in the context of this thesis (Strötgen et al., 2011; Strötgen and Gertz, 2012a), the extraction of spatio-temporal events is not a well-established task in the research community in contrast to, for instance, the task of temporal tagging addressed in Chapter 3. Thus, while annotation standards and annotated corpora are available for the temporal tagging task, neither annotation guidelines nor annotated corpora exist for spatio-temporal event extraction.

For the manual annotation of spatio-temporal events and also for the development of automatic extraction approaches that go beyond the cooccurrence approach, it is important to clearly specify when a combination of temporal and geographic expressions forms a valid spatio-temporal event. According to the definition of spatio-temporal events (Definition 4.1), we formulate the following specifications:

- A temporal expression te_i and a geographic expression ge_j have to cooccur within a window w in a document. This window w is set to one sentence.
- Within w, something has to be described which is, was, or will be happening at the time referred to by te_i at the place referred to by ge_j.
- The temporal expression has to be of type “date” or “time” since durations and set expressions often cannot be anchored on a timeline (cf. Section 2.3.2).

In Section 4.3.2, we already discussed examples of potential spatio-temporal events and explained when combinations of extracted temporal and extracted geographic expressions form valid events. All theses examples can be decided based on the four above described specifications.

Note, however, that it is important that the geographic and temporal expressions are indeed used to refer to the location and time of the spatio-temporal event, and that a location or time is not accidentally valid. For instance, in the *Spatio-temporal Event Example – Sentence 1*,[10] e_1 = ⟨"Sept. 20, 2011", "Greece"⟩ is not a valid spatio-temporal event although the valid event e_2 = ⟨"Sept. 20, 2011", "Athens"⟩ is geographically contained in it. Applying geographic mappings to the geographic component of e_2 thus transforms e_2 into e_1. However, the geographic containment relationship in the sentence between "Athens" and "Greece" is accidentally, and "Greece" could be replaced by any other geographic expression for which the containment relationship does not hold. If the sentence started with "Evangelos Venizelos, <PLACE>Turkey</PLACE>'s finance minister", it would be obvious that e'_1 = ⟨"Sept. 20, 2011", "Turkey"⟩ is not a valid spatio-temporal event since the sentence describes something happening in Athens and not in Turkey, i.e., the geographic expression "Turkey" – and thus "Greece" in the original example – is not used to refer to a location where a spatio-temporal event takes place.

Another difficult issue occurs when geographic expressions are not used to refer to a location but to another entity type like in the following example:

Spatio-temporal Event Example – Sentence 9.
In a historic first, <PLACE>*Iraq*</PLACE> *announced the creation of its first national park in* <TIME>*July 2013*</TIME>.
potential event e_{24} = ⟨"July 2013", "Iraq"⟩

In this example, it is described that a representative of the country "Iraq" announced something, i.e., "Iraq" is used to refer to an agent. Thus, "Iraq" is not used to refer to a location where a spatio-temporal event takes place and e_{24} is not a valid spatio-temporal event. However, it is quite obvious that one can also argue that the geographic expression is used to refer to an agent located in "Iraq", and thus, e_{24} could also be considered as a valid spatio-temporal event following the above annotation specifications.

A further reason making the above example and similar constructions difficult issues is that it is arguable if expressions in such contexts should be considered as geographic expressions at all and if they thus should be extracted by a geo-tagger. For instance, Leveling and Hartrumpf (2008) argue that expressions in these contexts should not be considered as locations since they are used metonymically and "[m]etonymic location names refer to other, related entities and possess a meaning different from the literal, geographic sense" (Leveling and Hartrumpf, 2008). Although it was shown that such expressions "are to be treated differently to improve performance of geographic information retrieval" (Leveling and Hartrumpf, 2008), most geo-taggers extract them.

Due to this conflict, we do not want to handle such potential events as invalid spatio-temporal events nor do we want to handle them in the same way as clearly valid events. In addition, spatio-temporal events can often be considered of taking place at the respective locations although the expressions are used metonymically – e.g., in "Washington announced", "Paris declines", and "Berlin says" where "Washington", "Paris", and "Berlin" refer to the governments located in the respective cities – so that we make the following distinction: We mark such events as "agent-based spatio-temporal events" when manually annotating spatio-temporal events in our annotated data sets, and allow their extraction when addressing the task automatically.

[10] For convenience, we repeat this sentence here: *Evangelos Venizelos, <PLACE> Greece</PLACE> 's finance minister, listens to an aide during a session of parliament in <PLACE> Athens</PLACE> on <TIME> Sept. 20, 2011</TIME> .*

Note that in the first example above, "Greece" cannot be considered as an agent since the agent is "Evangelos Venizelos, <PLACE>Greece</PLACE>'s finance minister". Thus, the event e_1 = ⟨"Sept. 20, 2011", "Greece"⟩ is neither a valid spatio-temporal event nor an agent-based spatio-temporal event.

4.5.3 Annotated Data Sets

For the development and evaluation of spatio-temporal event extraction methods that go beyond the cooccurrence approach, a development data set and an evaluation data set are required. As mentioned above, the development data set (the initial data set) is used to evaluate the cooccurrence approach, which allows for detecting specific patterns that can be used to develop the more complex approaches. These then cannot be evaluated on the initial data set so that further test data are required.

Document Selection

To decide what types of documents should be included in the data sets, the following requirements for the document selection are formulated:

(a) Temporal expressions have to be annotated.

(b) Geographic expressions have to be annotated.

(c) Spatio-temporal events have to be annotated.

(d) The documents should contain many temporal and geographic expressions as well as a reasonable amount of spatio-temporal events.

(e) Documents of different domains should be considered (according to the domain definition in Section 3.3.1, Definition 3.1, page 48).

(f) Documents of different languages should be included.

Obviously, in the context of developing and evaluating spatio-temporal event extraction approaches, (c) is the most important requirement. However, since there is no data set containing annotations of spatio-temporal events, requirements (a) and (b) become important because existing annotations of temporal and geographic expressions simplify the task of and reduce the effort for manually annotating spatio-temporal events. Unfortunately, there is no suitable corpus containing both, manually annotated temporal and geographic expressions. Thus, we build our data sets of some of the documents of the temporally annotated corpora described in Section 3.2.3. Using these corpora, requirement (d) is also considered, because these corpora all contain many temporal expressions, and, as detected during the annotation process, also several geographic expressions and spatio-temporal events.

In Chapter 3, we explained the different challenges for temporal tagging documents of different domains. Since the event-centric search and exploration scenarios that will be introduced in Chapter 6 should also be suitable for documents of different domains, we also want to evaluate the spatio-temporal event extraction approaches on different domains (requirement (e)). Thus, for English, we select documents of the WikiWars corpus and the TimeBank corpus, containing narrative- and news-style documents, respectively. Since both corpora are already annotated with temporal expressions, only geographic expressions and spatio-temporal events have to be manually annotated.

Finally, to satisfy requirement (f), we also select documents of the German WikiWarsDE corpus containing narrative-style documents. As for the English corpora, we manually annotated geographic expressions and spatio-temporal events. Although this data set is only used to evaluate the event extraction approaches, and although the more complex extraction methods are developed using an English data set only, we aim at developing language-independent event extraction methods. All event-centric search and exploration scenarios that will be developed in Chapter 6 are applicable on multilingual corpora.

Annotation Procedure

For the development and the evaluation of approaches to extract spatio-temporal events, it is not important to process full documents. In contrast, single sentences containing at least one temporal and one geographic expression are required. Thus, we run the following procedure to build the manually annotated data sets. Note that all documents are taken from temporally annotated corpora.

- Each document is split into sentences; sentences without date or time expressions are removed.
- All geographic expressions (toponyms) are manually annotated without normalization information.
- All sentences without at least one temporal and one geographic expression are removed.
- Some of the sentences – as will be detailed below – are randomly selected.
- Sentences are duplicated so that for each pair of temporal and geographic expressions, a separate sentence exists.
- In each sentence, the temporal expression of analysis and the geographic expression of analysis are marked as expressions of analysis to distinguish them from further occurring temporal and geographic expressions. Thus, each sentence instance contains a single cooccurrence of analysis.
- Following the annotation guidelines described in Section 4.5.2, each cooccurrence of analysis is manually annotated as (i) spatio-temporal event, (ii) agent-based spatio-temporal event, or (iii) no spatio-temporal event.

Except the number of sentences that are randomly selected, the same procedure is applied to all documents of the three corpora. As initial data set, we use 150 sentences of the WikiWars corpus. As evaluation data sets, we use 50 sentences of each of the three corpora. In Table 4.7, the four data sets are listed together with the number of cooccurrences in each set. In addition, an example sentence containing two temporal and two geographic expressions is shown in Table 4.8. Each cooccurrence with explicitly marked temporal and geographic expressions of analysis (TEA and GEA) is manually annotated.

Evaluation Results for the Cooccurrence Approach

As starting point for the development of more complex event extraction methods, we analyze all potential spatio-temporal events in the initial data set, i.e., the cooccurrences manually annotated as events, as agent-based events, or as non valid events. Obviously, using the cooccurrence approach all potential spatio-temporal events are extracted as spatio-temporal events and no distinction is made between clearly valid events and agent-based events. In Table 4.9, the respective evaluation numbers are presented.

Considering only clearly valid events, the precision is already above 50%. Combining the precision value with the cooccurrence approach's recall of 100%, results in an f_1-score (cf. Section 2.6.1) of 67.2%.

	development data set	evaluation data sets		
name	WW-150	WW-50	TB-50	WWde-50
corpus	WikiWars	WikiWars	TimeBank	WikiWarsDE
unique sentences	150	50	50	50
cooccurrences	411	111	91	102

Table 4.7: Development and evaluation data sets containing potential spatio-temporal events.

sentence with annotated expressions of analysis	manual annotation
German forces surrendered in <GEA>Italy</GEA> on <TEA>April 29</TEA> and in <G>Western Europe</G> on <T>May 7</T>.	event
German forces surrendered in <GEA>Italy</GEA> on <T>April 29</T> and in <G>Western Europe</G> on <TEA>May 7</TEA>.	no event
German forces surrendered in <G>Italy</G> on <TEA>April 29</TEA> and in <GEA>Western Europe</GEA> on <T>May 7</T>.	no event
German forces surrendered in <G>Italy</G> on <T>April 29</T> and in <GEA>Western Europe</GEA> on <TEA>May 7</TEA>.	event

Table 4.8: Manual event annotations; each cooccurrence is annotated separately.

When considering clearly valid and agent-based events as correct, the f_1-score even raises to 82.1%. Thus, the baseline for the evaluation of more complex approaches is already very strong.

Note that we also performed an evaluation of the cooccurrence approach in (Strötgen and Gertz, 2012a). There, however, we only used a data set containing Wikipedia articles. All sentences were taken of the WikiWars and WikiWarsDE corpora and contained manually annotated temporal expressions. However, for geographic expressions, we relied on automatic annotations of a geo-tagger. In contrast, as described above, we now use data sets of different domains for the development and evaluation of the further approaches and for a first evaluation of the cooccurrence approach. In addition, temporal expressions and geographic expressions are now manually annotated, and all cooccurrences of temporal and geographic expressions are manually checked for whether they form an event. Thus, this procedure allows for a better comparison of different approaches for spatio-temporal event extraction because errors of the geo-tagger and temporal tagger do not occur and thus do not influence the event extraction task.

Before describing some heuristic and linguistically-motivated approaches for spatio-temporal event extraction, note that in two student bachelor theses, preliminary advanced approaches for event extraction were studied. Kaufmann (2012) developed some heuristic and linguistically-motivated methods using manually created rules, and Limpert (2013) applied relation extraction methods followed by a machine learning post processing step. While these works are partially similar to the work we will present in the following sections, their evaluation data sets have the same deficits as the one we used for the evaluation described in (Strötgen and Gertz, 2012a). Furthermore, as mentioned above, we focus in this thesis on language-independent event extraction methods. Nevertheless, both works can be considered as helpful preliminary studies.

	cooccurrences	valid events	agent-based events	non-valid events	precision (valid)	precision (valid/agent-based)
WW-150	411	208	78	125	50.6%	69.6%

Table 4.9: Evaluation of the cooccurrence approach on the initial data set.

4.5.4 Heuristic Approaches for Event Extraction

In the following, some heuristic approaches will be explained. The key characteristic of these approaches is that they all make use of solely language-independent, easy-to-extract types of information.

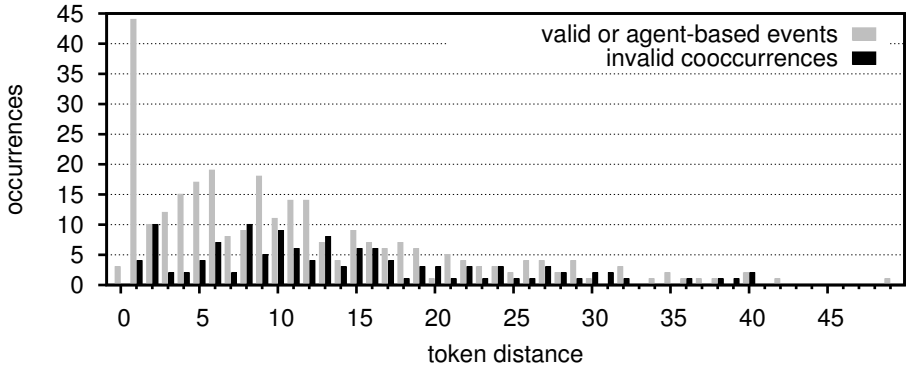
Token Distance

A first simple type of information is the word or token distance between the temporal expression of analysis and the geographic expression of analysis. For instance, in the example sentence provided in Table 4.8, the two cooccurrences with quite closely occurring temporal and geographic expressions are valid spatio-temporal events, while the other two cooccurrences are not valid events. Thus, a simple assumption that could be used to improve the precision of spatio-temporal event extraction is that the closer the temporal and the geographic expressions of a potential event occur in a sentence, the higher the probability that the two expressions form a valid spatio-temporal event.

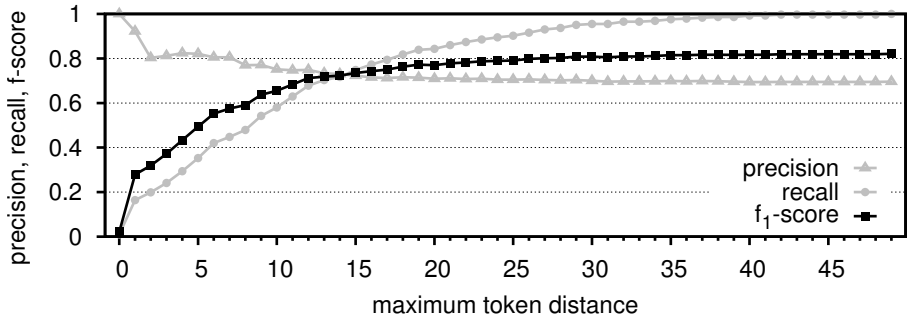
To determine if this assumption is valid and thus promising to improve spatio-temporal event extraction, we analyze all cooccurrences of the WW-150 data set. In Figure 4.5(a), we show the number of valid (or agent-based) events and the number of cooccurrences not forming valid events in relation to the token distance between the respective temporal and geographic expressions. As can be easily seen, in general, there are many more cooccurrences with rather small token distances between the two expressions of analysis. In addition, there are many more valid events with a token distance of one than for any other token distance (44). However, already at a token distance of two, the number of valid events equals the number of invalid events (10 each). Also, there are even more cooccurrences not forming a valid event than cooccurrences forming a valid event if the token distance equals eight. Nevertheless, even for large token distances, there are still many cooccurrences forming valid events and not only invalid ones – as one might have assumed.

In Figure 4.5(b), we show the precision, recall, and f_1-score values using each occurring token distance as the maximum allowed token distance between the temporal and geographic expressions of analysis. Obviously, allowing any token distance results in a recall of 100%, and, as mentioned above, in a precision of almost 70% (cf. Table 4.9). The resulting f_1-score of over 82% is already quite high and cannot be reached using any restriction of the token distance. However, if one is interested in a precision-optimized approach, one could set the maximum token distance to one for a precision of over 90% with a recall of about 16% or to seven for a precision of over 80% with a recall of almost 45%. However, it is obviously not possible to just set a token distance threshold to improve the event extraction with respect to the f_1-score.

In summary, limiting the token distance can help to improve the precision of spatio-temporal event extraction. However, the recall decreases dramatically. Thus, the token distance heuristic on its own is not suitable for high quality spatio-temporal event extraction. Below, we will study whether this heuristic in combination with other features is more valuable.



(a) Numbers of cooccurrences forming valid (or agend-based) and invalid events.



(b) Results of the cooccurrence approach with token distance thresholds.

Figure 4.5: Token distance statistics for cooccurrences in the WW-150 data set.

Number of Cooccurrences in a Sentences

Another idea to improve the event extraction quality is to exploit information about the number of cooccurrences in the sentences. Given a sentence with a single cooccurrence, it might be quite likely that this cooccurrence forms a valid event. In contrast, given a sentence with many cooccurrences, it might be likely that many of them do not form valid events. This information could either be used to reject all cooccurrences of sentences having many cooccurrences or to try to pick only cooccurrences that might be more likely valid events. In the following, we analyze the WW-150 corpus accordingly.

Single-Cooccurrence Sentences

Analyzing all sentences with a single cooccurrence independent of all other sentences, the following statistics arise. Instead of 411 cooccurrences, only 51 cooccurrences are under analysis, and 44 of them are valid (or agent-based) spatio-temporal events. Thus, when considering only single-cooccurrence sentences, the precision equals 86.3% with a recall of 100%. However, taking all 411 cooccurrences of the WW-150 data set into account with 286 valid (or agent-based) events, the recall equals only 15.4% with unchanged precision. Thus, this heuristic in isolation is again not suitable to optimize the f_1-score.

To analyze if token distance information can be used to improve event extraction from single-cooccurrence sentences, we show the number of valid (or agent-based) events and the number of cooccurrences not forming events in relation to the distance between the temporal and geographic expres-

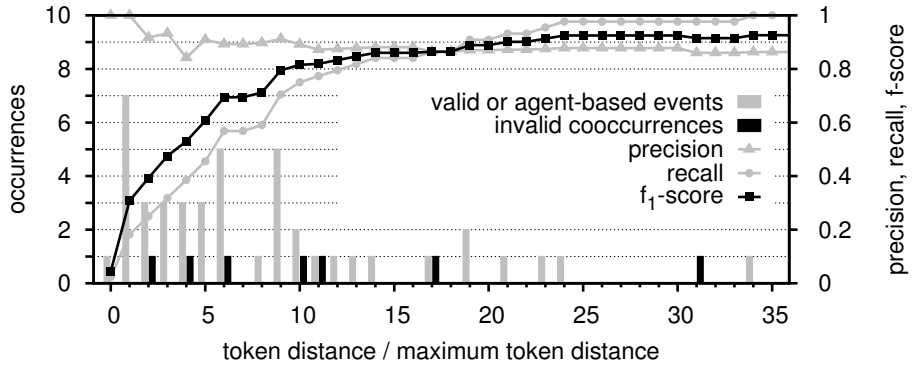


Figure 4.6: Token distance statistics for single-cooccurrence sentences in the WW-150 data set.

sions of analysis. In Figure 4.6, this distribution is shown together with the precision, recall, and f_1-score values considering all possible maximum token distances. Note that these values are based on the 51 cooccurrences. Obviously, the recall and the f_1-score are much lower when considering the full data set.

In summary, considering only single-cooccurrence sentences, the precision of the event extraction process can be increased from less than 70% to over 86%. However, the other cooccurrences should be analyzed separately because otherwise the recall of the event extraction drops dramatically. Considering token distance information, the precision can be slightly increased further to 90% but only at the cost of a decreasing recall from 100% to 70.5%. Since the precision without limiting the maximum token distance is already very high, token distance information alone does again not help to optimize the f_1-score.

Sentences with Multiple Cooccurrences

When considering only sentences with multiple cooccurrences, several strategies can be applied to improve the precision of the event extraction process. First, one can set a constraint that each temporal and geographic expression is only extracted as part of a single event using the closest expression of the other type not already extracted as part of another event. Second, one can set a constraint that a temporal and a geographic expression are only extracted as event if no other temporal or geographic expressions occur between the two expressions of analysis. Third, one can set a constraint that each temporal and geographic expression have to be extracted as part of an event using the closest expression of the other type independent of whether it is already extracted as part of another event.

In the following, we analyze these three strategies based on all sentences of the WW-150 data set that contain multiple cooccurrences. Since there were 51 single-cooccurrence sentences in the data set, the remaining 99 sentences contain at least two cooccurrences and a total of 360 cooccurrences. Of these 360, 242 form a valid (or agent-based) event while 118 do not. Thus, the simple cooccurrence approach on this subset results in a precision of about 67%, a recall of 100%, and an f_1-score of 80.4%.

At Most One Event per Expression

As first method for extracting spatio-temporal events from sentences with multiple cooccurrences, we specify that each temporal and each geographic expression can only be extracted as part of a single event. Thus, there might be temporal and geographic expressions that are not extracted as part of any event

at all. For instance, assuming a sentence that contains two geographic expressions and one temporal expression, this method extracts the temporal expression with the closer occurring geographic expression as spatio-temporal event, while the other geographic expression is not extracted. In the example shown in Table 4.8, the two valid events are correctly extracted by the method, while the two invalid cooccurrences are correctly not extracted.

In the WW-150 data set, of the 360 cooccurrences, 135 are extracted as spatio-temporal events by this method, and 114 of them are valid spatio-temporal events. Obviously, this method has a rather low recall and a rather high precision, with 47.1% and 84.4%, respectively. The f_1-score thus equals 60.5% when considering the 360 cooccurrences. Below, we combine the methods to extract events from sentences with multiple cooccurrences with the single-cooccurrence sentence approach and present evaluation numbers on the full WW-150 data set for a meaningful comparison among the three methods and in relation to the cooccurrence approach.

No Expressions between Expressions of Analysis

As second method, we specify that a temporal and a geographic expression are only extracted as spatio-temporal event if there is no other temporal or geographic expression between them. In the WW-150 data set, there are 159 such cooccurrences. Of these 159 extracted events, only 116 cooccurrences form a valid event while 43 do not. Thus, considering again all 360 cooccurrences of the sentences with multiple cooccurrences of which 242 are valid events, the precision equals 73.0%, the recall is about 47.9%, and the f_1-score thus equals only 57.9%.

With respect to the example presented in Table 4.8, this method extracts the same two cooccurrences as events as the first method. However, the third listed cooccurrence is incorrectly extracted, too. Assuming again a sentence with two geographic expressions and one temporal expression, the number of extracted events depends on the sentence structure. If the temporal expression occurs between the geographic expressions, both cooccurrences are extracted as events. Otherwise, only one event is extracted.

Comparing the first and the second method with each other, we can see that the first method has a much better precision (11.4 percentage points). Although the recall of the second method is slightly better than the one of the first method, it is still rather low. Thus, while the motivation for the second method was to extract more events if the decision between possible extractions is difficult, the goal of significantly increasing the recall without worsening the precision compared to the previous method was not reached. Given the recall of 100% when using the simple cooccurrence approach, we try to delimit the decrease of the recall in addition to improving the precision with the third method.

Each Expression as Part of an Event

As third method, we specify that each temporal and geographic expression has to be part of at least one event. For this, we extract events according to the following strategy. For each expression of one type, we select the closest expression of the other type in the sentence to form an event. Given the example in Table 4.8, this methods again correctly extracts both valid events while both invalid cooccurrences are not extracted. However, in contrast to the first method, a radical change occurs with respect to extracting events of sentences where the number of temporal and geographic expressions is unbalanced. Assuming a sentence with a single temporal expression but multiple geographic expressions, then each geographic expression forms an event with the temporal expression, i.e., many more events are extracted.

	true pos.	false pos.	false neg.	precision	recall	f_1-score
cooccurrence approach	286	125	0	69.6	100.0	82.1
single-cooccurrence sentences (scs)	44	7	242	86.3	15.4	26.1
at most 1 event per expression + scs	158	28	128	84.9	55.2	66.9
no other expressions in between + scs	160	50	126	76.2	55.9	64.5
each expression part of event + scs	252	79	34	76.1	88.1	81.7

Table 4.10: Combining single-cooccurrence sentence and multi-cooccurrence sentence methods.

With this approach, 280 cooccurrences are extracted as spatio-temporal events with 208 of them being valid spatio-temporal events. Thus, although the precision is only 74.3% and thus ten percentage points lower than with the first approach, the recall equals 86% and remains quite high. Thus, an f_1-score of 79.7% on the subset of sentences with multiple cooccurrences is reached, which is almost identical to the f_1-score of the pure cooccurrence approach.

Examples for which this strategy works particularly well are enumerations as in the following sentence, in which both cooccurrences form valid events: *"This was followed up with simultaneous raids against anarchists in <G>Petrograd</G> and <G>Moscow</G> at <T>the end of April</T>"*. Obviously, enumerations and temporal or geographic expressions forming groups (e.g., *"between <T>April</T> and <T>May</T>"*) could also be extracted in a preprocessing step so that such expressions can be handled as a single expression independent of the extraction method. However, since the identification of enumerations and syntactic groups is linguistically-motivated and requires at least some language knowledge, it will be covered in Section 4.5.5.

Combining Single-Cooccurrence Sentence and Multi-Cooccurrence Sentence Methods

In addition to extracting events with these three methods from sentences with multiple cooccurrences, all cooccurrences of single-cooccurrence sentences can be extracted as events as described above. In Table 4.10, we show the evaluation numbers of the three approaches together with the single-cooccurrence sentence approach on the full WW-150 data set to compare the results with the simple cooccurrence approach. Obviously, using any of the three heuristics improves the precision of the event extraction process. However, only the third method can almost equalize the loss in recall.

4.5.5 Linguistically-motivated Approaches for Event Extraction

In the following, we present linguistically-motivated approaches to improve the precision of the event extraction process. First, we exploit simple syntactic structures such as sub-sentences and enumerations. Then, we combine these methods with the heuristic approaches. Finally, we briefly discuss the value of using part-of-speech and dependency parsing information for spatio-temporal event extraction.

Splitting Sentences into Sub-sentences

A very simple and language-independent method to split a sentence into sub-sentences is to rely on the use of semicolons. Doing so, and handling all resulting sub-sentences in the same way as standard sentences results in the following changes: Instead of a total of 411 cooccurrences, the WW-150 data set then contains

(a) Patterns for geographic enumerations and groups.

pattern	frequency	pattern	frequency
<P> (and|or) (the)? <P>	16	<P> in (the)? <P>	5
<P> , <P>	4	<P> , ((the)? <P>)+ ,? (and|or (the)? <P>	4
<P> (\()? near (the)? <P>	2	<P> , (the)? <P> , ((the)? <P>)+	1
<P> district of (the)? <P>	1	<P> (and|or) <P> in (the)? <P>	1

(b) Patterns for temporal enumerations and groups.

pattern	frequency	pattern	frequency	pattern	frequency
<T> (-|/) <T>	6	<T> to <T>	5	<T> (and|or) <T>	5

Table 4.11: Patterns for geographic (a) and temporal (b) enumerations and further syntactic groups.

only 392 cooccurrences. However, out of the 19 eliminated cooccurrences, four valid spatio-temporal events are also removed. When relying on sub-sentence information, these four cooccurrences cannot be extracted independent of the applied method for event extraction.

However, when using sub-sentence information, the number of single-cooccurrence sentences increases from 51 to 55, and the number of valid events increases from 44 to 50. Thus, not only the recall is better when using sub-sentence information with the single-cooccurrence approach (from 15.4% to 17.5%), but also the precision (from 86.3% to 90.9%). An example where sub-sentence information is helpful is the *Spatio-temporal Event Example – Sentence 8* (page 135) where all five cooccurrences not forming events can be correctly excluded from the set of extracted events.

Enumerations and Further Syntactic Groups of Temporal or Geographic Expressions

As explained in the context of the third heuristic to extract events from sentences with multiple cooccurrences, enumerations or other syntactic groups of temporal or geographic expressions frequently occur in the sentences of the WW-150 data set. Thus, we try to extract such enumerations and syntactic groups in a preprocessing step to exploit them for improving the precision of the event extraction process.

Based on all sentences in the WW-150 data set, we craft several simple patterns to extract enumerations and further syntactic groups. These pattern are depicted in Table 4.11. This group information is then used in two scenarios: (i) All sentences containing at most one group of temporal and one group of geographic expressions are handled as single-cooccurrence sentences, and all respective cooccurrences are extracted as spatio-temporal events. (ii) Each group of temporal or geographic expressions is handled as single expression and the above explained heuristics are again evaluated and compared with each other.

Handling all sentences with at most one temporal and one geographic enumeration or group as single-cooccurrence sentences, the number of extracted events significantly increases from 51 to 108. Furthermore, 96 of the extracted events are valid events so that the recall raises from 15.4% to 33.6% and the precision from 86.3% to 88.9%. Exploiting sub-sentence information additionally, 112 cooccurrences are extracted as events and 102 of them are also valid. Thus, precision and recall can be further improved.

Note that extracting enumerations and further syntactic groups of temporal and geographic expressions is a language-dependent task. However, the patterns are rather simple and can be translated without

	true pos.	false pos.	false neg.	precision	recall	f_1-score
cooccurrence approach	286	125	0	69.6	100.0	**82.1**
+ sub-sentences	282	110	4	71.9	98.6	**83.2**
single-cooccurrence sentences (scs)	44	7	242	86.3	15.4	26.1
+ sub-sentences	50	5	236	90.9	17.5	29.3
+ enumerations	96	12	190	88.9	33.6	48.7
+ sub-sentences + enumerations	102	10	184	91.1	35.7	51.3
at most 1 event per expression + scs	158	28	128	84.9	55.2	66.9
+ sub-sentences	161	23	125	87.5	56.3	68.5
+ enumerations	204	34	82	85.7	71.3	77.9
+ sub-sentences + enumerations	208	27	78	88.5	72.7	79.8
no other expressions in between + scs	160	50	126	76.2	55.9	64.5
+ sub-sentences	160	45	126	78.0	55.9	65.2
+ enumerations	212	57	74	78.8	74.1	76.4
+ sub-sentences + enumerations	212	50	74	80.9	74.1	77.4
each expression part of event + scs	252	79	34	76.1	88.1	81.7
+ sub-sentences	254	68	32	78.9	88.8	**83.6**
+ enumerations	262	81	24	76.4	91.6	**83.3**
+ sub-sentences + enumerations	264	71	22	78.8	92.3	**85.0**

Table 4.12: Comparing evaluation results of the cooccurrence approach, the heuristic approaches, and the simple linguistically-motivated approaches, and their combinations on the WW-150 data set. F_1-scores of cooccurrence approaches and of methods outperforming them are highlighted.

much effort. In addition, no further language-dependent NLP tool is necessary so that the event extraction process can be performed in any language as long as a sentence splitter, a temporal tagger, and a geo-tagger for the respective language are available.

Combining Multiple Approaches

In Table 4.12, we show evaluation results for the cooccurrence approach (with and without the sub-sentence feature) and for combining the heuristic approaches described in Section 4.5.4 with the sub-sentence and enumeration approaches. While adding one type of information already improves each of the heuristic approaches with respect to precision and recall, adding both types of information further improves both values. Using the third heuristic approach, i.e., that each expression has to be part of an event and thus forms an event with the closest occurring expression of the other type, the cooccurrence approach is outperformed with respect to the f_1-score on the WW-150 data set.

More Sophisticated Linguistic Approaches to Spatio-temporal Event Extraction

Obviously, the heuristic and simple linguistically-motivated approaches described so far do not take into account any deeper natural language processing techniques. Nevertheless, the evaluation results on the WW-150 data set are good with values of about 79%, 92%, and 85% for precision, recall, and f_1-score, respectively. However, when analyzing the incorrectly extracted events, we detected two main types of errors that both can probably only be addressed when taking into account deeper natural language

processing information. In the following, we discuss the value of softening the language-independence requirement for the event extraction process and to apply deeper natural language processing techniques.

The first type of errors are due to geographic expressions used as modifiers that do thus not refer to locations where something is happening. Most errors, however, occur because temporal and geographic expressions do not syntactically belong together, e.g., one of the expressions occurs in a subordinated clause that is unrelated to the other expression. While the first error type can be addressed using part-of-speech context information, the latter can be addressed exploiting dependency parsing information.

Exploiting Part-of-Speech Context Information

Two examples of the first error type that were already described in Section 4.3.2 when analyzing cooccurrences of three example documents are *"Evangelos Venizelos, <GEA>Greece</GEA>'s finance minister..."* and *"The <GEA>EU</GEA> statement said ..."*. To address such errors, part-of-speech context information might be helpful to exclude geographic expressions occurring as modifiers in a noun phrase from the set of geographic expressions forming events.

We thus experimented with several part-of-speech context patterns to detect geographic expressions occurring as modifiers. However, excluding all extracted events with such geographic expressions worsened the evaluation results on the WW-150 data set since many more valid than invalid events were excluded. Some of the incorrectly excluded events are agent-based events with phrases such as *"After <GEA>Moscow</GEA>'s Bolshevik government ..."*. While this has been partially expected, many more incorrectly excluded spatio-temporal events are even regular valid spatio-temporal events. Examples are *"Confederate incursions into <GEA>New Mexico</GEA> territory were repulsed in <TEA>1862</TEA> ..."*, *" ... arrived at the <GEA>Bagram</GEA> Air Base on <TEA>July 7</TEA>"*, and *"... after the <POI>Paris</POI> Peace Accords were signed in <TOI>1973</TOI>."*.

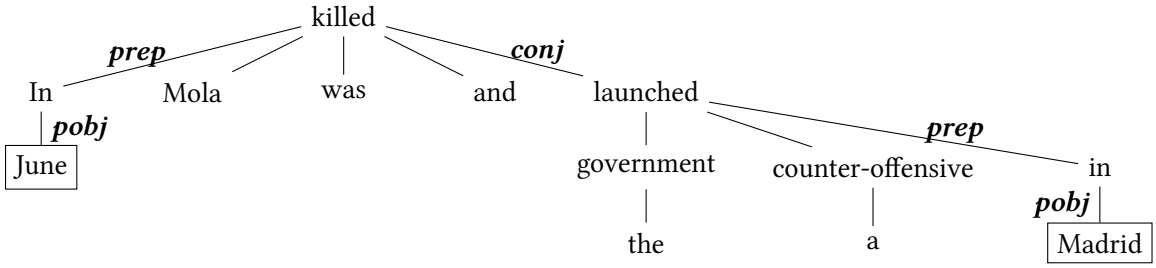
Unfortunately, the part-of-speech context patterns of geographic expressions contributing to valid or agent-based events and part-of-speech context patterns of geographic expressions forming only invalid events are identical. That is, the same patterns lead to correctly and to incorrectly excluded geographic expressions. Thus, we do not include an event extraction approach that is based on part-of-speech context information to try to improve the precision of the event extraction process.

Note that some of the errors of the part-of-speech context patterns could be avoided when including semantic or lexical information in addition to part-of-speech information, e.g., to detect that "territory" and "Air Base" refer to locations. However, using this type of information would make the event extraction approach even more language-dependent. Since our goal in this thesis is to exploit extracted event information from multilingual corpora, we aim at a high quality event extraction process that is as language-independent as possible.

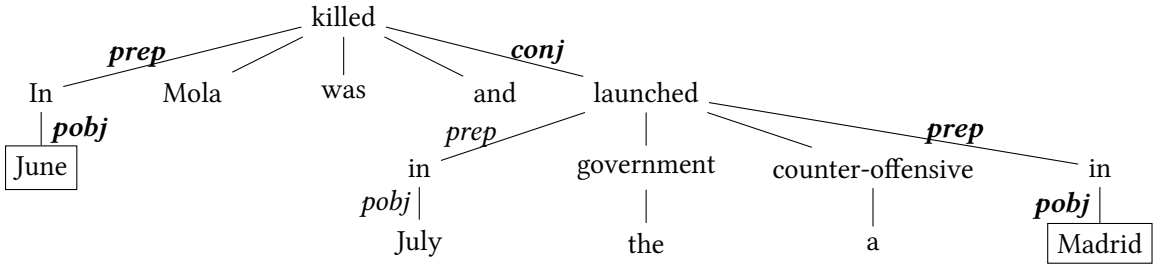
Exploiting Dependency Parsing Information

The second error type – responsible for most of the incorrectly extracted events when using the third heuristic approach in combination with the sub-sentence and group features – cannot be addressed using part-of-speech context patterns since the errors occur due to syntactic reasons.

A promising way to address these errors is to analyze the syntactic structure of the sentences containing cooccurrences. In general, the task of relation extraction is often addressed using dependency parsing



(a) In June, Mola was killed, and the government launched a counter-offensive in Madrid.



(b) In June, Mola was killed, and, in July, the government launched a counter-offensive in Madrid.

Figure 4.7: Example for the difficulties of exploiting dependency parsing information for spatio-temporal event extraction. The dependency parses between the temporal and the geographic expressions are identical in (a) and (b), but the two expressions only form a spatio-temporal event in (a). The sentences are processed with the Stanford Dependency Parser (de Marneffe et al., 2006).

information. For instance, in the BioNLP domain, there are such relation extraction approaches, e.g., for extracting protein-protein interactions from text documents (see, e.g., Fundel et al., 2007). Typically, such approaches analyze the dependency parses between two entities to decide whether a relationship exists.

However, exploiting dependency parsing information for spatio-temporal event extraction is often more difficult than for other types of relation extraction such as protein-protein interactions. In the latter case, entities are typically part of complements, i.e., obligatory parts of a sentence as the subject or a required object. In contrast, temporal and geographic expressions are often part of adjuncts, i.e., optional parts of a sentence. Thus, it is often not possible to decide whether a temporal and a geographic expression form an event solely based on the dependency parse between the two expressions. For instance, in both sentences presented in Figure 4.7, the dependency parses between "June" and "Madrid" are identical. However, only in the first sentence, the two expressions form an event. In the second sentence, nothing is described that happened at the respective point in time ("June") at the respective location ("Madrid").

That exploiting dependency parsing information for spatio-temporal event extraction is quite challenging was also reported by Kaufmann (2012). He developed some approaches based on dependency parsing information to extract spatio-temporal events and to distinguish between agent-based and standard valid events. While these approaches worked quite well for distinguishing agent-based events from other valid events, they hardly increased the event extraction performance compared to much simpler methods and the cooccurrence approach.

	WW-150			WW-50			TB-50			WWde-50		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
cooc. approach	69.6	100	**82.1**	73.9	100	**85.0**	68.1	100	**81.0**	72.5	100	**84.1**
+ sub-sentences	71.9	98.6	**83.2**	76.6	100	**86.8**	68.1	100	**81.0**	74.7	100	**85.5**
single-cooc. sent. (scs)	86.3	15.4	26.1	90.5	23.2	36.9	87.0	32.3	47.1	92.3	32.4	48.0
+ sub-sentences	90.9	17.5	29.3	90.9	24.4	38.5	87.0	32.3	47.1	93.1	36.5	52.4
+ enumerations	88.9	33.6	48.7	90.9	36.6	52.2	90.6	46.8	61.7	95.5	56.8	71.2
+ sub-sent+ enum.	91.1	35.7	51.3	91.2	37.8	53.4	90.6	46.8	61.7	95.7	60.8	74.4
at most 1 event + scs	84.9	55.2	66.9	87.3	58.5	70.1	78.8	66.1	71.9	91.4	71.6	80.3
+ sub-sentences	87.5	56.3	68.5	87.3	58.5	70.1	78.8	66.1	71.9	91.4	71.6	80.3
+ enumerations	85.7	71.3	77.9	87.7	69.5	77.6	80.7	74.2	77.3	94.0	85.1	**89.3**
+ sub-sent+ enum.	88.5	72.7	79.8	87.7	69.5	77.6	80.7	74.2	77.3	94.0	85.1	**89.3**
no exp. between + scs	76.2	55.9	64.5	77.1	57.3	65.7	74.2	74.2	74.2	87.1	73.0	79.4
+ sub-sentences	78.0	55.9	65.2	78.3	57.3	66.2	74.2	74.2	74.2	87.1	73.0	79.4
+ enumerations	78.8	74.1	76.4	76.0	69.5	72.6	76.5	83.9	80.0	87.6	86.5	**87.1**
+ sub-sent. + enum.	80.9	74.1	77.4	77.0	69.5	73.1	76.5	83.9	80.0	87.6	86.5	**87.1**
each exp. event + scs	76.1	88.1	81.7	81.6	97.6	**88.9**	69.0	96.8	80.5	89.0	98.6	**93.6**
+ sub-sentences	78.9	88.8	**83.6**	81.6	97.6	**88.9**	69.0	96.8	80.5	90.1	98.6	**94.2**
+ enumerations	76.4	91.6	**83.3**	82.0	100	**90.1**	69.0	96.8	80.5	89.0	98.6	**93.6**
+ sub-sent. + enum.	78.8	92.3	**85.0**	82.0	100	**90.1**	69.0	96.8	80.5	90.1	98.6	**94.2**

Table 4.13: Event extraction evaluation results of all approaches on the development and test sets. The f_1-scores of the cooccurrence approaches and of the methods outperforming them are highlighted.

Finally, to exploit dependency parsing information, additional language-dependent tools would have to be used so that the event extraction process becomes further less language-independent. Having the goal of high quality event extraction from multilingual corpora in mind, deep natural language processing tools are rather counter-productive. Thus, and since the much simpler above presented approaches already result in high quality event extraction on the WW-150 data set, we do not further study event extraction approaches exploiting dependency parsing information in this thesis.

4.5.6 Evaluation and Comparison

All heuristic and linguistically-motivated approaches have been developed based on the analysis of the cooccurrences available in the WW-150 data set. In this section, we evaluate their performance on unseen data, namely the three evaluation data sets described in Section 4.5.3.

In Table 4.13, the evaluation results for all approaches and combinations are given. On all data sets, the cooccurrence approach achieves an f_1-score of over 80%. Thus, independent of the domain and language of the data sets – English narrative (WW-150, WW-50), German narrative (WWde-50) or English news (TB-50) – this simple approach can already be used for high quality event extraction.

When considering not only single-cooccurrence sentences, the first heuristic approach (at most one event per expression) outperforms the second heuristic approach (no other expressions in between). The best f_1-score values of the three heuristic approaches are achieved with the third heuristic approach (each expressions forms an event with the closest occurring expression of the other type).

Combining the heuristic approach with the simple linguistically-motivated approaches leads to better results in most cases. An exception is the news data set on which the sub-sentence feature does not improve the results with any of the heuristic approaches. In general, the best results with respect to the f_1-score are achieved with the third heuristic approach in combination with both simple linguistically-motivated features although the cooccurrence method outperforms this approach on the news data set. While the improvement of this third heuristic with sub-sentence and enumeration features over the cooccurrence approach was moderate on the development data set (three percentage points), it is higher on the WW-50 data set (5 percentage points) and even much higher on the WWde-50 data set (10 percentage points).

With respect to the usability of the approaches, all heuristic approaches and the sub-sentence feature are easily applicable without any language-dependent adaptations. In contrast, the enumeration feature has to be adapted to each language. Although only a few simple patterns have to be translated, at least some language knowledge is required to transfer the approach from one language to another. However, for none of the approaches, deeper language-dependent NLP tools are required.

In summary, the cooccurrence approach is the simplest approach for event extraction and already achieves high evaluation results, and can be further improved using the sub-sentence feature. In particular when being interested in recall-optimized event extraction, this approach is to be recommended. When being interested in high precision event extraction, the first heuristic approach outperforms the other approaches. If language-specific adaptations are possible, it can be combined with sub-sentence and enumeration information and achieves on all data sets a recall of at least almost 70%. Finally, the best results with respect to the f_1-score are achieved by the third heuristic in particular when combined with the two simple linguistically-motivated approaches. Then, the precision can be improved compared to the cooccurrence approach with only slight decreases of the recall.

4.5.7 Summary

In this section, we developed heuristic and simple linguistically-motivated approaches for spatio-temporal event extraction and compared them with the cooccurrence approach. In addition, we discussed the value of using language-dependent, deep natural language processing techniques to improve the event extraction quality. However, already simple approaches and even the cooccurrence approach achieve high quality evaluation results so that the need for language adaptations and language-dependent tools is limited when being interested in event extraction from multilingual corpora. This is particularly true if a high recall in the event extraction process is desirable.

Distinguishing between clearly valid and agent-based spatio-temporal events is also possible (cf. Kaufmann, 2012). However, since this differentiation is not required for our event-centric exploration scenarios that will be developed in Chapter 6, we did not present any methods for this task in detail.

4.6 Summary of the Chapter

In many text documents, events play an important role. Motivated by the key characteristic of events, i.e., that they occur at some specific time and some specific location, we introduced in this chapter the concept of *spatio-temporal events*. As surveyed at the beginning of this chapter, event concepts exist in many research areas so that a differentiation to other concepts was necessary.

In addition to defining *extracted temporal expressions* and *extracted geographic expressions*, we defined *spatio-temporal events* in a quite simplistic but precise way as combinations of temporal and geographic information extracted from text documents. Based on several examples, we explained when combinations of temporal and geographic expressions form spatio-temporal events.

Due to the simplicity of spatio-temporal events, i.e., that we solely rely on temporal and geographic information, the key characteristics of temporal and geographic information are inherited to events. Thus, not only temporal and geographic expressions but also events are well-defined, can be normalized, and can be organized hierarchically. These key characteristics make events term- and language-independent and allow to concisely "calculate" with events. For this, we developed several methods to compare temporal or geographic expressions with each other and to map them to be equal, which will be used to measure the similarity between events. These methods can either be applied to events or to extracted temporal and geographic expressions independent of whether they are part of an event. Thus, these methods as well as the concepts of temporal, geographic, and event document profiles form the basis for the following chapters where we develop several spatio-temporal and event-centric search and exploration tasks.

In addition to the theoretical aspects of events, we also addressed the task of spatio-temporal event extraction from text documents. Having in mind that the event-centric search and exploration scenarios shall be applicable to multilingual document collections, we focused on language-independent methods to extract events. By developing heuristic and simple linguistically-motivated approaches, we showed that high quality spatio-temporal event extraction can be reached already by applying simple extraction strategies. Our evaluation on data sets of different domains and languages further demonstrated that this fact is generalizable across text domains and languages.

Strictly speaking, our spatio-temporal events – independent of the extraction method – consist only of information about the temporal and geographic components of the events. However, since the document and offset information for the two expressions forming an event are also available, the contexts from which the events are extracted are also directly accessible. In addition, using our approach of spatio-temporal events, it is also possible to associate persons with events and to build personalized event profiles. For this, each extracted spatio-temporal event is associated with each person that is mentioned in the same sentence as the event. Applying named entity tools to detect person names and cross-document coreference resolution tools to assign person names to real world entities, each person mentioned in a corpus can be associated with its events. As will be described in Chapter 6, personalized event profiles can be exploited to combine event-centric and person-centric exploration scenarios, e.g., to detect event-centric person similarity.

5 Spatio-temporal Information Retrieval

In this chapter, we address the topic of spatio-temporal information retrieval, i.e., information retrieval for satisfying diverse temporal and geographic information needs. For this, we first outline the importance of this topic and formulate the objectives of this chapter in Section 5.1. In Section 5.2, we give an overview of the literature related to temporal, geographic, and spatio-temporal information retrieval.

In Section 5.3, we present our approach to multidimensional querying, which allows to combine standard textual queries with temporal and geographic constraints. Then, we introduce our proximity^2-aware ranking model for textual, temporal, and geographic queries taking into account two types of proximity measures in addition to addressing the three query dimensions. While the theoretical model is developed in Section 5.4, we explain some indexing and querying details in Section 5.5, and evaluate our approach in Section 5.6. Finally, we summarize the chapter in Section 5.7.

5.1 Motivation and Objectives

In many types of documents, temporal and geographic information plays a pivotal role – as it was already exemplarily shown in Figure 4.1 (page 133) for three types of documents. In addition, temporal and geographic information needs are quite frequent and important in many search scenarios. For example, Metzler et al. (2009) report that 7% of the queries in an analyzed query log have an implicit temporal intent, and the query log analysis of Nunes et al. (2008) reveals that 1.5% of Web queries contain explicit temporal information. Note that this was even an underestimated number since their evaluation of the used temporal tagger on a subset of the analyzed query log data showed a low recall of only 63% (Nunes et al., 2008). Similarly, Zhang et al. (2006) attest the frequency of geographic information needs by reporting that 12.7% of the queries in an analyzed query log contain some kind of geographic information.

There probably would be even more search queries with explicit temporal and geographic information needs if there were better ways to properly query documents whose content is constrained to specific time intervals or geographic regions. However, there are two shortcomings when one is faced with temporal and geographic information needs. (i) In standard search engines, all information needs have to be expressed by words and there are no other ways to specify geographic and temporal parts of a query. (ii) Geographic and temporal expressions are usually treated in the same way as regular terms – not only in term-based queries but also in the texts of the document collection. Thus, their meaning cannot be exploited to satisfy respective information needs. In summary, geographic and temporal information needs are very frequent but not well served by standard search engines.

This issue of standard search engines is further supported by the example presented in Figure 5.1. For a typical spatio-temporal information need (*"world records between 1965 and 1974 in Central Europe"*), we queried three major standard search engines (Google, Yahoo!, and Bing) using the verbatim information need as search query ((a), (c), and (e)). Obviously, these search engines are not tailored for specifying geographic or temporal constraints about the content of documents so that we used the textual description

(a) Google's top three results (verbatim query).

(b) Google's top four results (adapted query).*

(c) Yahoo!'s top three results (verbatim query).

(d) Yahoo!'s top three results (adapted query).

(e) Bing's top three results (verbatim query).

(f) Bing's top three results (adapted query).

Figure 5.1: Screenshots of Google's, Yahoo!'s, and Bing's top three search results for our spatio-temporal information need *world records between 1965 and 1974 in Central Europe*. In (a), (c), and (e) for the verbatim query; in (b), (d), and (f) for the query *"world records" 1965-1974 Central Europe*. * Google lists as best results the extended version of our paper (Strötgen and Gertz, 2013b) containing this example. Thus, we show three further search results.
Sources: `http://www.google.com`, `http://www.yahoo.com`, `http://www.bing.com` [last accessed August 17, 2014].

of our information need. Although we do not know a lot about the strategies of these search engines to answer our queries, the snippets help us to get an impression of which parts of the information need were understood correctly and which parts caused difficulties.

For instance, neither any temporal expression referring to a date within the query time interval nor any geographic expression referring to a location within the query region is highlighted except the terms used to specify the boundaries of the time interval and to describe the query region. Furthermore, some result pages describe some kind of world records but no result page contains a hint about any world record between 1965 to 1974 in Central Europe.[1] In contrast, if we queried for *"'world records' Munich 1972"*, all three search engines deliver valuable results about some world records that occurred during the 1972 summer Olympics in Munich since these documents all contain the words "Munich" and "1972" that can be directly matched with the query terms.[2] However, if we are faced with our initial information need, we obviously do not want to list all (sport) events that took place in Central Europe between 1965 and 1974 where world records might have occurred to receive valuable search results.

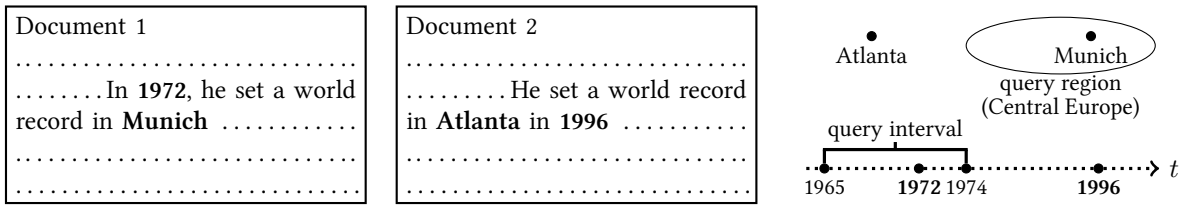
Having information needs such as *"world records between 1965 and 1974 in Central Europe"*, one is mainly faced with two problems: (i) the time interval and the geographic region have to be identified as such kinds of information, and (ii) it has to be verified for all temporal and geographic expressions in the documents if they belong to the queried interval and region. If temporal and geographic expressions are not identified and normalized, a search engine cannot assign different relevance scores to similar but different documents as those depicted in Figure 5.2(a) and Figure 5.2(b). However, the two documents should be treated differently. Thus, the temporal and geographic expressions in the query and the documents have to be identified and the temporal and geographic knowledge about these expressions, as depicted in Figure 5.2(c), has to be exploited.

Note that the temporal expressions in the example documents are explicit temporal expressions, and that the described issues become even more problematic if underspecified and relative temporal expressions (cf. Section 2.3.2) such as *"September"* or *"ten years later"* occur in documents. Assuming that a query interval would be correctly understood, such expressions could not be validated if they were not normalized.

Our goal in this chapter is to address the shortcomings of standard search engines related to spatio-temporal information retrieval. For this, we first survey related work and then model approaches for multidimensional querying to combine textual queries with temporal and geographic constraints. Furthermore, we develop a ranking model that takes into account all three query dimensions and combines them in meaningful ways by exploiting the key characteristics of temporal and geographic information.

[1] An exception is the second document in Figure 5.1(a) *Sport in Poland*, which lists the sprinter Irena Szewińska under "Famous Polish athletes" and explains that "[b]etween 1964 and 1980 Szewińska participated in five Olympic Games" and that "[s]he also broke six world records". However, it is not clear if any of the world records happened in Central Europe between 1965 and 1974. A second search result with partially valuable information is the second document in Figure 5.1(c), containing a link to the Wikipedia page "World records in athletics" `http://en.wikipedia.org/wiki/List_of_world_records_in_athletics`.

[2] For instance, `http://www.olympic.org/munich-1972-summer-olympics` occurs in the top three results of all three search engines. Further top three results are, e.g., Wikipedia pages (`http://en.wikipedia.org/wiki/1972_Summer_Olympics`, `.../wiki/Mark_Spitz` and `.../wiki/Swimming_at_the_1972_Summer_Olympics`).



(a) First example document. (b) Second example document. (c) Relevant knowledge.

Figure 5.2: Two example documents with temporal and geographic expressions in (a) and (b), as well as helpful temporal and geographic knowledge about occurring expressions to answer the example information need *"world records between 1965 and 1974 in Central Europe"* in (c).

5.2 State-of-the-Art in Temporal, Geographic, and Spatio-temporal Information Retrieval

In this section, we survey related work on spatio-temporal information retrieval. Since temporal information retrieval (TIR) and geographic information retrieval (GIR) are often considered separately, we first discuss relevant approaches to TIR (Section 5.2.1) and GIR (Section 5.2.2) before surveying related work addressing both dimensions in Section 5.2.3.

A further related research topic is entity-oriented search. There, temporal and geographic information is also sometimes studied (see, e.g., Hu et al., 2006), but usually the focus lies on other named entity types such as persons and products. While the task is quite similar to TIR and GIR in one aspect – "[i]n entity-oriented search, identifying named entities in documents as well as in queries is the first step towards high relevance of search results" (Jiang, 2012: p.16) – a major difference is that in entity-oriented search, search results are typically a ranked list of entities of the queried type rather than documents (Balog et al., 2012). Thus, we do not detail any approaches to entity-oriented search in this section, but briefly refer to it in Chapter 6 where we develop our event-centric search and exploration scenarios.

5.2.1 Temporal Information Retrieval

In general, the research area of temporal information retrieval covers several sub-topics. Although the sub-topics partially overlap, we present in the following some example approaches clustered according to what kind of temporal information is involved.

Time as Dimension of Relevance

As a first aspect, time can be a dimension of relevance in addition to topical relevance. In this case, the creation time of the documents can be exploited. For instance, a user may favor recent documents for specific news-related queries so that the freshness of search results may be important (Li and Croft, 2003). Note, however, that generally preferring recent documents degrades performance for non recency-sensitive queries, so that search results can be improved when automatically identifying queries, which prefer recent documents (Dai et al., 2011; Efron and Golovchinsky, 2011). More generally, time intervals of interest for specific queries may be determined to improve the ranking of search results by mainly selecting documents of the respective interval (Dakka et al., 2008, 2012), in particular when querying news archives. Another example application is time-based review analysis (Strötgen et al., 2012a).

Time as Context Information

As a second aspect, time can be important as context information. Identical queries may represent different information needs depending on when they were formulated so that understanding temporal query dynamics is crucial. In their study of a large-scale query log, Kulkarni et al. (2011) determine different changes in query popularity such as periodicities, trends, and spikes, which can be exploited to determine the users' query intents. In addition to improving search results, time as contextual information can be used to perform time-sensitive query auto-completion (Sengstock and Gertz, 2011; Shokouhi and Radinsky, 2012). In general, time-sensitivity can be determined on different levels of granularities, e.g., there are seasonal queries (Shokouhi, 2011), queries with peaks on a particular day of a week, and those depending on the time of the day (Beitzel et al., 2004).

Applications Exploiting Extracted Temporal Information

Finally, there are several sub-topics that exploit temporal information occurring in the documents. For this, temporal expressions are extracted from the texts and normalized (cf. Chapter 3). Based on the normalized information, search and exploration tasks can be performed (Alonso et al., 2007), as we surveyed focusing on research trends and challenges (Alonso et al., 2011). How to handle temporal information in Web search engines was also discussed from the database perspective by Manica et al. (2012).

Besides applications such as timeline clustering of general search results (Alonso et al., 2009), exploration of news documents along a timeline (Matthews et al., 2010), temporal summaries of news topics (Allan et al., 2001), temporal question answering (Pasca, 2008), or searching and exploring statements about the future (Baeza-Yates, 2005; Jatowt et al., 2009; Dias et al., 2011), an important sub-topic considers time as a query topic. Since we are addressing mainly this last aspect of temporal information retrieval in this chapter, we now focus on describing related approaches to time as query topic.

Time as Query Topic

Time can either occur explicitly or implicitly as query topic. According to Jones and Diaz (2007), queries without explicit temporal information can either be atemporal (e.g., "poaching"), temporally unambiguous (e.g., "Battle of Gettysburg"), or temporally ambiguous (e.g., "Iraq war"). An example of addressing implicit temporal queries is suggested by Metzler et al. (2009). Given a query without temporal information, it is validated whether the query is an implicit temporal query. For this, query log analysis is performed to search for identical queries with explicit year information. If an implicit temporal query is detected, the initial search results retrieved for the text query are re-ranked by boosting the relevance scores of those documents, in which the determined year expressions occur (Metzler et al., 2009).

While Metzler et al.'s approach requires query log analysis and only allows for determining year information, some approaches on implicit temporal queries do not exploit information about the temporal expressions in the documents at all. For instance, Kanhabua and Nørvåg (2010) suggest three approaches to determine implicit query times, with two of them relying on a temporal language model for the query terms and all terms of top-k retrieved documents, respectively, while the third approach assumes the document creation times of the top-k retrieved documents as the time of interest. Then, the search results are re-ranked by boosting "documents with creation times that closely match with the [determined query] time" (Kanhabua and Nørvåg, 2010). Thus, such approaches could be considered as approaches using time as dimension of relevance rather than as query topic.

Finally, there are also works that exploit temporal information extracted from the documents' content to determine the implicit time information of a query, e.g., by using Web snippets to date time-implicit queries (Campos et al., 2011, 2012). In this approach, date expressions are extracted from the Web snippets of the top-k relevant documents retrieved for a text query. Then, similarity scores between the query and candidate date expressions are calculated to identify those date expressions that are most relevant for the respective query (Campos et al., 2012).

While the first step when dealing with implicit temporal queries is to detect the implicit query time, this step is obviously not required when processing queries with explicit time information. Approaches on explicit temporal queries allow the user to explicitly specify time intervals that the documents should be about, i.e., all such approaches exploit temporal information extracted from the documents' content. Since we also focus on explicitly formulated temporal (and geographic) information needs in this chapter, we present in the following approaches addressing explicit temporal queries.

Addressing Explicit Temporal Information Needs

When processing explicit temporal queries, a temporal and a topical score are usually calculated for the temporal and textual parts of a query, respectively, to calculate a total relevance score. Note that not the document creation time or the time of the last modification of a document is relevant as it is characteristic for standard search engines, but the temporal content of the documents is analyzed. Thus, the temporal score can either be determined by considering all temporal expressions in the documents of a document collection, or by validating a temporal focus time or primary time of a document (see, e.g., Strötgen et al., 2012b; Jatowt et al., 2013), which can be determined independent of a query. In the next paragraphs, some approaches are briefly explained following the one strategy or the other.

Jin et al. (2008)

One of the early works describing a search engine for combining textual and temporal constraints is TISE, a time-inspired search engine for Chinese Web content (Jin et al., 2008). Processing two query parts – a textual query and a temporal query specified as time interval – a final ranking is calculated by combining a page importance score, a text similarity score, and a temporal similarity score. While the importance score and the text similarity score are independent of the temporal information need, the temporal score is determined based on comparing the query interval and the so-called "primary time" of the document. This primary time is detected for each document independent of the queries and identified in a rule-based way (e.g., dates appearing in the title are treated as primary times). Thus, all other temporal expressions in the documents are not considered for calculating the temporal relevance score (Jin et al., 2008).

Vicente-Diez and Martinez (2009)

A temporal Web search engine for Spanish is proposed by Vicente-Diez and Martinez (2009). They extract and normalize temporal expressions in both, documents and queries, and replace the expressions by their normalized values. While explicit, relative, and underspecified temporal expressions are supported, only time point expressions are considered. Thus, they can rely on a standard ranking measure because they handle the normalized temporal expressions in the same way as standard query terms. Obviously, only documents containing exactly the same time point as specified in the query can be determined as being temporally relevant.

Arikan et al. (2009) and Berberich et al. (2010)

Similar to Jin et al. (2008), Arikan et al. (2009) assume that a query can be split into a textual and a temporal part, with the temporal component being "a set of temporal expressions" (Arikan et al., 2009). However, in contrast to Jin et al., they do not only consider one major time (or time interval) for each document, but all extracted temporal expressions occurring in the content of the documents. Then, for their ranking approach, they distinguish between regular terms and temporal expressions and create two language models for the two kinds of information. Given a query, the probabilities that the textual content and the temporal content of a document are generated by the language models are calculated. However, they "assume that the generation of the textual query [...] and the temporal query [...] happen independently" (Arikan et al., 2009). Despite that weakness of their approach – which we will further analyze in Section 5.4 when presenting our spatio-temporal ranking model – their evaluation results clearly demonstrate the importance of carefully handling temporal information in information retrieval.

Another minor weakness of their approach occurs probably due to the lack of sophisticated, publicly available temporal taggers at the time of their work. Instead of extracting all types of temporal expressions, they only extract simple explicit temporal expressions from the documents that match "against a set of regular expressions capturing common formats of temporal expressions" (Arikan et al., 2009). Thus, relative and underspecified expressions are not considered by their approach (cf. Chapter 3).

This first approach to integrate extracted temporal expressions into a language model was extended by Berberich et al. (2010). While already Arikan et al. considered each temporal expression as an interval with a begin and an end boundary, the extended approach takes care of the uncertainty of temporal expressions, i.e., that "it is not clear which exact time interval they [(temporal expressions)] actually refer to" (Berberich et al., 2010). As motivating examples, the phrases "in 1998 Bill Clinton was President of the United States" and "France won the FIFA World Cup in 1998" are presented. Obviously, the first refers to the whole year and the second to a specific day – although the identical temporal expression is used. Thus, instead of using single begin and end boundaries, each temporal expression is now represented as a four-tuple describing the lower and upper bounds of the begin and end boundaries of a time interval (Berberich et al., 2010). A further extension to the work of Arikan et al. (2009) is that broad experiments were conducted on the New York Times corpus. The used queries were formulated and the relevance assessments were collected by performing user studies. As in the original approach, the results show the importance of not treating temporal expressions as regular terms, and additionally, that taking into account the uncertainty of temporal expressions further improves the ranking results.

Kanhabua and Nørvåg (2012)

Kanhabua and Nørvåg (2012) also assume explicitly provided temporal queries in their time-aware learning to rank-based approach. By combining temporal- with entity-based features, they do not only treat temporal expressions in a special way but also other types of named entities, e.g., persons and locations. Their temporal features to determine the temporal similarity between queries and the documents consider both, the temporal expressions in the documents' texts and document creation time information with and without concerned uncertainty. The entity features are used to calculate the semantic similarity. For ranking documents, they use the weighted sum of the feature scores, i.e., make the same independence assumption between the textual and the temporal query parts as the above described approaches.

In their experiments on the New York Times corpus using the queries and relevance judgments of Berberich et al. (2010), they outperform the approach of Berberich et al. (2010). As can be validated thanks

to their extensive experiments on the importance of the single features (Kanhabua and Nørvåg, 2012), the better ranking results are mainly due to some of the entity-based features and the temporal features considering the document creation time. Note, however, that the experiments are performed on a news corpus for which the document creation time plays a crucial role in general. In addition, they benefit from the types of the queries since many of them contain named entities in the textual components of the queries (cf. Berberich et al., 2010).

Indexing Temporal Information

Another important research question in temporal information retrieval is how to deal with temporal information so that it can be accessed efficiently. While there is some work on how to store dynamic content, i.e., temporally versioned document collections such as Web archives (e.g., Berberich et al., 2007), another issue is how to store temporal information extracted from the content of the documents for efficient access. Obviously, we do not deal with dynamic content in our work but with the latter issue.

Vicente-Diez and Martinez (2009) just added the normalized values of temporal expressions to a regular inverted index. Obviously, this strategy is only possible when dealing with time points only and when the time points of a query have to match exactly a time point in the documents. Once time intervals are supported, this strategy cannot be applied anymore. Since only the so-called primary time of a document is supported by Jin et al.'s approach, they can make use of a hybrid temporal text index, "which groups primary time and text key words into one uniform index structure" (Jin et al., 2008). However, such an index becomes unfeasible once more than a single temporal expression or interval is associated with a document. Thus, a standard way to handle textual and temporal information from the documents' content is to create two types of indexes (e.g., Arikan et al., 2009). Finally, Berberich et al. suggest to "keep track of the documents that contain a specific temporal expression [and to organize] [i]ts lexicon, which consists of temporal expressions [...] using interval trees" (Berberich et al., 2010).

Summary

Temporal information retrieval covers several sub-topics.[3] The sub-topic we are focusing on in this chapter are explicitly expressed temporal information needs. To satisfy such information needs, the content of the documents that are to be queried have to be processed by a temporal tagger to make available normalized information about occurring temporal expressions. While the described approaches to satisfy explicit temporal information needs vary in the way they address this research question, they all share the same weakness, namely that the textual and the temporal parts of a query are considered as being independent. This issue will be tackled in our spatio-temporal information retrieval model.

5.2.2 Geographic Information Retrieval

There are different points of view on what kind of information geographic information retrieval (GIR) deals with. For instance, following Usery (1996), in the context of digital libraries it is sometimes assumed that "Geographic Information gathers three dimensions, namely spatial, temporal, and topical" (Palacio et al., 2010). However, we regard GIR similar as Jones and Purves and as in the general GIR research context observed, e.g., in the GIR workshop series,[4] namely that GIR "is concerned with the problems of

[3] A new survey paper on temporal information retrieval was published very recently (Campos et al., 2014).

[4] Proceedings of the workshop series on geographic information retrieval: GIR'05 (Jones and Purves, 2005), GIR'07 (Purves and Jones, 2007), GIR'08 (Jones and Purves, 2008), GIR'10 (Purves et al., 2010), GIR'13 (Jones and Purves, 2013).

finding information resources that relate to particular geographic locations" (Jones and Purves, 2006). In Section 5.2.3, we will discuss related work in spatio-temporal information retrieval, separately.

Similar as temporal information retrieval, GIR covers several sup-topics. Since we mainly address explicit geographic (and temporal) information needs in this chapter, we focus on this issue. However, we also briefly present some of the other sub-topics to allow for a more complete overview of geographic information retrieval.

Note that it is important to distinguish different kinds of geographic location information that can be associated to documents, e.g., Amitay et al. (2004) distinguish source geography (the origin, physical location of the server, address of the author) and target geography of a Web page which relates to the content and topic of the page. Since the target geography highly depends on the geographic information mentioned on the page, one can also distinguish three types of geographic information, as done by Wang et al. (2005a,b) who named them provider location, serving location, and content location.

Geographic Information as Context Information

Considering geographic context information – the user's location – can help to better understand a user's information need, because identical queries may represent different information needs depending on where they were formulated. Thus, they require different serving locations in the results. However, there are several challenges that need to be addressed.

Obviously, the first challenge is to decide whether a query targets locally relevant or globally relevant pages, i.e., if this context information may be relevant for a given query at all. To tackle this issue, Gravano et al. (2003), for instance, use a variety of machine-learning techniques to determine the (implicit) geographic locality of queries by categorizing them as local or global. Once a query is identified as being local, detecting the location of interest is the next challenge. Note, however, that one has to distinguish between local queries for which the location of interest corresponds to the location where the query was formulated and those local queries that contain an implicit location of interest. How to address the latter types of queries is discussed below when surveying research on geographic information as query topic. Examples of location-based queries (location of interest equals location where the query was formulated) are typical local search queries, e.g., as those asking for restaurants or shops.

In contrast to determining the time when a query is formulated, determining where it is formulated is often more difficult. Consequently, to perform local search or location-aware search, there is some work on geolocating Web queries. While mobile devices send their current geographic location as metadata and thus directly allow the exploitation of their location information (Mountain and MacFarlane, 2007), non-mobile devices can sometimes be tracked by their IP addresses. For instance, Backstrom et al. (2008) describe a probabilistic framework to localize queries of a query log. By assigning "locations to a (large) subset of the IP addresses issuing the queries, [...] [they] define the geographic focus of a topic by the locations of the people who search for it" (Backstrom et al., 2008).

Independent of whether a query is determined as global or local, and whether the local information corresponds to the user's location or to some implicit geographic information in the query, it is important to know about the geographic scope of potentially relevant documents, as will be detailed next.

Geographic Scope Detection

A popular research area in geographic information retrieval is to detect the geographic scope of Web documents. This task is often addressed by analyzing place references occurring in the documents. One of the pioneering works in this area is the GIPSY system (Woodruff and Plaunt, 1994). After geographic expressions are extracted and disambiguated to their polygons, all overlaps of the polygons are summarized to find the most specific regions for each document.

Ding et al. (2000) – an extended work of Buyukkokten et al. (1999) – propose two methods for assigning geographic scopes to Web documents. The first one exploits the geographical distribution of hyperlinks to the resources while the second approach relies on the textual content of the documents. Furthermore, they introduce the measures "power" and "spread" for determining the importance of detected locations and for specifying the geographic extent of the geographic scope of a document, respectively. Thus, a geographic scope can be of different granularities, e.g., a city or a country.

Similar to the GIPSY system, the Web-a-Where system (Amitay et al., 2004) aims at extracting and normalizing all geographic expressions from documents and at calculating the geographic focus of a document by applying the following strategy: (i) each occurring geographic expression "adds a certain score to the importance of this place [...] [and] lower scores to the enclosing hierarchies" (Amitay et al., 2004), e.g., a city expression weights for itself and with a lower score to the state and the country in which it is located; (ii) the regions with the highest importance scores can be determined as focus or foci of a document. Thus, while some approaches assign a single geographic scope to a document, others, as the Web-a-Where system, allow several relevant foci for a document. If multiple foci are to be determined, the task is also sometimes referred to as assigning geographic signatures to documents (Batista et al., 2010).

Further research on this topic includes the approaches by Wang et al. (2005a), who analyze content, hyperlink, and user log information, and Silva et al. (2006) who employ a graph-ranking algorithm that exploits ontology information of geographic concepts. In addition, Anastácio et al. (2009) empirically compare four different approaches and three baselines with geographic scores assigned by humans on a collection of 6,000 Web pages. More recently, Cheng et al. (2010) proposed an approach to geo-locate Twitter users by solely exploiting the content of the user's tweets.

Geography as Query Topic

A sub-topic of geographic information retrieval for which geographic scope information is often exploited is geography as query topic. Geography information can either occur explicitly or implicitly as query topic. As mentioned above, queries without explicit geographic location information can either be global or local, i.e., they can either contain or not contain an implicit geographic information need. In local queries, the location information of interest can either be the user's location or some other implicit location information.

Obviously, information about geographic scopes of the documents can be helpful to boost those documents that satisfy the geographic information need, or, in the case of global queries, to boost documents without geographic scopes or scopes of rather coarse granularities. Thus, most of the works described above motivate the task of geographic scope detection by its value to improve information retrieval.

Geographic-aware Search Engines

In the following, we present some systems that do not only aim at detecting geographic scopes but at improving geographic-aware information retrieval. For instance, Ding et al. (2000) developed a geographic-aware search engine for newspapers. After estimating a geographic scope for each document as described above, the user is asked to enter a ZIP code in addition to the text query. The system then first ranks the documents according to a standard text search, filters out all documents with scopes not covering the specified ZIP code area, and finally re-ranks the results by combining the textual score and the geographic scope score (Ding et al., 2000).

Two systems addressing the full GIR task including the extraction of geographic expressions, the (explicit) geographic querying, the ranking and the visualization of the search results are STEWARD (Lieberman et al., 2007) and SPIRIT (Purves et al., 2007). In the following, we exemplarily present details of these systems, before finally briefly discussing some further geographic-aware search engines.

The STEWARD System

The STEWARD system (Lieberman et al., 2007) considers all the steps required by a geographic-aware search engine. After document retrieval and standardization, the full task of geographic tagging is performed, i.e., the extraction of toponyms and their disambiguation, which is mainly based on the assumption that ambiguous geographic entities are the more likely, the more evidence they give to other geographic entities in the same document. Once all geographic entities are disambiguated, the geographic scope is computed by considering both, the geographic proximities of geographic entities and their contextual proximities in the documents.

For addressing the retrieval task, it is assumed that keywords are stored in an inverted index and the geographic scope of each document in a spatial index. Based on these indexes, STEWARD answers queries containing a geographic part, a keyword part, or both. If only a geographic part is available, the documents "are ranked by the extent to which STEWARD determines that the geographic entity in the query serves as the geographic focus of the document" (Lieberman et al., 2007). In the case of a simple keyword search, a standard text ranking is performed. However, "all of the references to geographic locations in each document [are also identified and ranked] [...] in the order in which it determines that they serve as the geographic focus of the document" (Lieberman et al., 2007). For combined queries, a boolean keyword search is performed and documents "are ranked in increasing order of distance of their geographic focus from the geographic location component of the query string" (Lieberman et al., 2007). Thus, the focus lies on the geographic component since no textual ranking is considered.

STEWARD's user interface contains a pane for textually specifying the geographic and textual parts of a query. In addition, a map can be used to select the location of interest. For visualizing the search results, a ranked list of documents is presented, and icons are placed on the map at the positions of the documents' geographic focus locations.

The SPIRIT System

Similar to STEWARD, the SPIRIT system is also a complete solution to geographic information retrieval (Purves et al., 2007). The approach to extract and normalize geographic expressions is less sophisticated, namely "a simple gazetteer lookup approach [...] [combined] with context rules and

additional name lists" (Purves et al., 2007) to detect non-location names is used for the extraction, and "a default sense approach and global geographical world knowledge" (Purves et al., 2007) to resolve ambiguities. However, one of the main contributions is that spatial relationships in queries are considered, i.e., not only the typically addressed "inside" relation is covered, but also further relationships such as "near", "south of", and "within distance of". Thus, triplets of the form <theme> <spatial relationship> <location> are handled as queries as an alternative to map-based formulated queries.

Note that the SPIRIT system can be considered as the outcome of the SPIRIT (Spatially aware Information Retrieval on the Internet) project, a European Commission funded project with several collaborators (Jones et al., 2002). Thus, a rather detailed requirements analysis has been performed and several of the identified requirements for the formulation of queries, for the nature of results, and for the user interface have been addressed. While some aspects will be discussed later (e.g., query interfaces), we briefly describe in the following how SPIRIT indexes spatial and thematic content of documents, how it determines the geographic relevance, and how it combines thematic and geographic relevance scores.

For keyword search, SPIRIT relies on an inverted index. Bounding box information of the disambiguated geographic expressions are used to create a geographic footprint for each document, and for geographic indexing, "a regular grid-based spatial indexing scheme [...] [is used dividing] the entire footprint coverage of the document collection into a grid of rows and columns" (Purves et al., 2007). Then, a list of document IDs is associated with each cell of the grid. For the final indexing, three approaches are suggested: (i) independent spatial and thematic indexes, (ii) an inverted index for each cell of the spatial index grid, and (iii) an extended inverted index so that the documents "are grouped according to the spatial index cells to which they relate" (Purves et al., 2007). Obviously, (ii) and (iii) result in an indexing overhead that increases with the number of footprints associated with the documents, but both are faster for retrieving search results than independent indexes.

The determination of thematically and spatially relevant documents for a query depends on the spatial relationship used in the query. In general, for each document, a spatial similarity score and a textual similarity score to the query are calculated and then combined into a single score. The spatial score relies either on the containment relationship between the query's and the document's footprints (inside relation), or on the proximity of the centroids of the query's and the document's footprints (near relation), or on the angle additionally (direction relations). Then, the standard text score and the spatial score are both normalized into the range [0,1]. Finally, "the documents [are ranked] in ascending order of their Euclidean distance from point (1,1) that is assumed to be the most relevant possible document" (Purves et al., 2007).

Although, Purves et al. (2007) point out the difficulties of evaluating spatially-aware search engines, they present system- and user-centered evaluations showing, e.g., that map-based querying can be intuitive, but also that "the system's overall precision could be considered as rather low [...] [mainly because] the number of georeferenced documents is relatively small [...] [and because] not all documents are correctly georeferenced" (Purves et al., 2007).

Further Geographic-aware Systems

Markowetz et al. (2005) describe a prototype of a geographic search engine for the ".de"-domain. Similar to the above described approaches, it relies on geographic footprints. These are calculated based on features extracted from the content of the documents (only zip codes, telephone numbers, and town

names), their URLs, and the WHOIS entries of the Web pages. Furthermore, due to empty footprints for many pages, "forward and backward propagation [is performed] across links as well as between co-cited pages" (Markowetz et al., 2005) based on the assumption that "[i]f one page has a geographic footprint, then a page it is linked to is [...] likely to be relevant to the same or a similar region" (Markowetz et al., 2005). As in the STEWARD system, the geographic queries are evaluated against a raster-based index (quad-tree). For ranking search results, the "weighted sum of its term-based score, its geographic score, and maybe an additional measure such as Pagerank" (Markowetz et al., 2005) is calculated. However, no evaluation results are reported.

Silva et al. (2006) use their graph-based scope detection approach briefly mentioned above to incorporate geographic search into a prototype Web search engine for Portuguese documents (Geo-Tumba). Geographic queries can either be textually formulated or using a map. In the case of ambiguous textual queries, the user interface helps to resolve the geographic scope of the query so that "it is submitted to the search engine and a list of the most relevant pages with scopes matching the query is returned" (Silva et al., 2006). That is, the geographic scope serves as a filter for textually relevant documents.

Further Ranking Approaches to Address Geographic Queries

A particular challenge for answering geographic information needs is how to combine geographic and thematic relevance measures. As surveyed above, typical methods are to use one of the dimensions for ranking and the other dimension as a filter, to use the weighted sum of individual scores, or other query independent measures such as the product or the maximum of individual scores. In contrast, Yu and Cai (2007) suggest to weight the dimensions dynamically, i.e., to use a query-aware ranking method, by "measur[ing] the relative importance of thematic and geographic relevance through analyzing [...] how specific (or general) a query is" (Yu and Cai, 2007).

Another approach to avoid a heuristic combination of individual relevance scores is to learn a ranking function. Martins and Calado (2010) present such a learning to rank approach for geographic information retrieval using the GeoCLEF datasets, which will be described below. Based on a set of textual features (standard IR ranking measures), geographic features (several similarity measures proposed in the GIR literature based on distance and/or containment information), and average features (heuristic combinations of textual and geographic features), they use the SVM^{map} framework for optimizing the mean average precision (cf. Section 2.6.2). All geographic features are based on the geographic scopes of the documents, but it is pointed out that "[i]t would be interesting to experiment with features computed from the individual placenames mentioned in the documents" (Martins and Calado, 2010). Despite difficulties such as the different characteristics of the topics (queries) in the GeoCLEF data sets, promising results were achieved and the approach "outperforms previous approaches based on heuristic combinations of features" (Martins and Calado, 2010).

Furthermore, there are few systems that do not only combine textual and spatial search but also allow for adding temporal constraints. These will be presented in Section 5.2.3 when discussing related research on spatio-temporal information retrieval.

The GeoCLEF Evaluation Campaign

GeoCLEF is an evaluation campaign for geographic information retrieval and aims at "provid[ing] the necessary framework in which to evaluate GIR systems for search tasks involving both spatial and

multilingual aspects" (Mandl et al., 2009). After a pilot project in 2005 (Gey et al., 2006), three further tracks were organized in 2006, 2007, and 2008 (Gey et al., 2007; Mandl et al., 2008, 2009).[5] All GeoCLEF tracks use large corpora containing news-style documents, and for each track, 25 queries (so-called topics) were defined, with each topic containing some kind of geographic component. In addition, relevance assessments were provided by the organizers resulting in a total of 100 topics with relevance assessments.

Note that "one goal of GeoCLEF is the creation of a geographically challenging topic set" (Mandl et al., 2009) so that difficulties were added on purpose, e.g., "vague geographic regions [...] geographical relations beyond IN [...] granularity below the country level [...] terms which do not occur in documents" (Mandl et al., 2009). In addition, proximity (e.g., "Most visited sights in the capital of France and its vicinity"), inclusion (e.g., "Attacks in Japanese subways"), and exclusion (e.g., "Portuguese immigrant communities around the world") issues were included. Nevertheless, neither in the document collection nor in the topics any kind of markup of geographic entities was provided, and "[s]ystems were expected to reveal that information b[y] themselves from the topic which resembles a more realistic task" (Mandl et al., 2009). While this aspect might be realistic when having standard search engines in mind, it is contradictory to the geographic-aware search engine approaches developed outside of GeoCLEF, which all aim at providing useful query interfaces for specifying geographic constraints. Thus, GeoCLEF can at least be partially regarded as question answering task considering geographic aspects rather than a pure GIR task.

In general, the GeoCLEF evaluation campaign further improved the popularity of addressing geographic queries. A variety of approaches has been developed and tested by the participants "ranging from basic IR approaches to deep natural language processing" (Mandl et al., 2009). Several approaches combined standard IR techniques "with similarity metrics for geographic scopes based on distance and/or containment" (Martins and Calado, 2010). For instance, in 2008, the best system for English was an ontology-based approach (Wang and Neumann, 2009) but surprisingly, a standard IR system not considering any geographic reasoning or knowledge source at all achieved the best results for several subtasks (Larson, 2009b). A reason is probably that processing the query, and in particular understanding the geographic intent, was already quite challenging.

In addition, to the GeoCLEF tasks, GikiP and GikiCLEF were organized in 2008 (pilot project) and in 2009.[6] The goal was to soften "the hard boundaries between question answering (QA) and information retrieval (IR)" (Santos and Cardoso, 2008) by evaluating "searches for Wikipedia entries which require some geographical processing" (Mandl et al., 2009). Topics were defined as questions, and the results were expected to be lists of links to Wikipedia pages (in any language) linking to documents about the correct type. For instance, each result page for the topic "Which Swiss cantons border Germany?" had to be about a canton itself not another page also containing an answer to the question (Santos et al., 2008). A further goal was to address cross-lingual and cross-cultural issues, i.e., to provide "questions about which one would expect a particular language or culture to display far more information than others" (Santos and Cabral, 2010). In retrospect, the answer type constraint was considered as one of the major flaws of GikiCLEF besides "a tremendous bias towards English" (Santos et al., 2010). In conclusion, GikiP and GikiCLEF are even more different from the GIR search engine approaches presented above than GeoCLEF.

[5] The websites of the four single events are: `http://ir.shef.ac.uk/geoclef/2005/`, `http://ir.shef.ac.uk/geoclef/2006/`, `http://ir.shef.ac.uk/geoclef/2007/`, and `http://www.uni-hildesheim.de/geoclef/` [last accessed August 14, 2014].

[6] The websites of GikiP and GikiCLEF are: `http://www.linguateca.pt/GikiP/` and `http://www.linguateca.pt/GikiCLEF/` [last accessed August 14, 2014].

GIR and Digital Libraries

In general, geographic information plays an important role in digital libraries and some applications addressing geographic information retrieval have been suggested (e.g., Larson and Frontiera, 2004; Larson, 2009a). As mentioned above, in this context, it is sometimes assumed that geographic information does not only combine the spatial and topical dimensions but also the temporal dimension (Palacio et al., 2010). Thus, we describe Palacio et al.'s approach to determine the effectiveness of GIR systems in digital libraries and their MIDR 2010 test collection in Section 5.2.3.

Geographic Querying

Another important aspect in GIR is geographic querying. In contrast to formulating temporal queries, specifying a geographic information need is not that straightforward. Typically, temporal information needs can be expressed as a boolean combination of time intervals by mentioning interval boundaries explicitly. In contrast, geographic information needs often cannot be specified with words mainly due to the following two reasons: (i) not every arbitrary region can be referred to with a name, and (ii) specifying boundaries of regions, e.g., with latitude/longitude information, is not common. Not even rectangular regions can typically be referred to with latitude/longitude information by humans, but polygonal regions obviously not at all.

Nevertheless, a textual description of a geographic information need is one of two frequently suggested methods in addition to map-based solutions. For instance, Geo-Tumba (Silva et al., 2006) uses a text field to specify the geographic query, GIPSY (Woodruff and Plaunt, 1994) provides a map interface, and SPIRIT (Purves et al., 2007) and STEWARD (Lieberman et al., 2007) contain both. If a user has to describe a geographic information need textually, it is usually expected that location names are used. With this method, all kinds of administrative regions can be specified. However, note that even frequently used region names often do not have fix boundaries, e.g., names of neighborhoods or colloquial region names such as "Southern Germany". Furthermore, due to ambiguity issues of location names, a system should either provide help in disambiguating ambiguous names (e.g., Silva et al., 2006), or an automatic disambiguation is a potential error source. In case of an incorrect disambiguation, it is almost impossible to successfully answer the information needs.

The second type of approaches are map-based, and also one of the requirements for GIR search engines formulated by Purves et al. is that "[i]t should be possible for users to specify the area of interest on a map" (Purves et al., 2007). Already Woodruff and Plaunt stated that a "map-based graphical interface has several advantages over a modal which uses text terms and over a model which uses numerical access to coordinates" (Woodruff and Plaunt, 1994). One advantage is that all kinds of regions can be specified even those that cannot be referred to with a name. While most of the map-based approaches support the specification of rectangles, Kumar et al. (2013) suggest an approach that goes beyond that. They recommend – for general geographic querying, i.e., not only for typical geographic search engines – "to provide users the ability to arbitrarily define their own spatial region of interest" (Kumar et al., 2013), because text-based inputs restrict search to predefined boundaries.

Applications Exploiting Extracted Geographic Information

As for temporal information, there are also further types of applications that exploit geographic information extracted from the content of documents. One example is the NewsStand system (Teitler et al., 2008).

It "monitors RSS feeds from thousands of online news sources, [...] extracts geographic content from articles" (Teitler et al., 2008), clusters the news stories, and a sophisticated map-based user interface allows for a deep geography-centric exploration of news content. Instead of RSS feeds, TwitterStand (Sankaranarayanan et al., 2009) uses tweets to detect late-breaking news. Thus, not only news articles but also opinions about the news can be explored geographically.

Geographic Indexing

A final important research question in GIR is how to index geographic information to guarantee fast query processing. In general, textual data is usually stored in an inverted index containing an entry for each (normalized, stemmed, or lemmatized) word and a postings list with references to the documents for each entry (for details see, e.g., Manning et al., 2008: p.6). Note that while adding each extracted geographic expression in a normalized form to such an inverted index makes it possible to search for locations, all further geographic information is lost, e.g., hierarchy information cannot be captured. Thus, to determine the relevance of documents with respect to the thematic and the geographic dimension, textual and geographic information is either indexed separately or in a hybrid or combined index.

In the case of separate indexing, there are several multidimensional index structures for spatial information such as R-trees, quad-trees, k-d-trees, but also grid indexes and space filling curves (Martins et al., 2005). An overview of multidimensional indexes for point and region data with detailed explanations is presented by Gaede and Günther (1998). Very popular spatial index methods are the R-tree and its numerous extensions since "[a]n R-Tree efficiently supports operations such as enclosure [...], intersection [...], nearest neighbor [...] and closest pairs [...] [, i.e., the operations which] form the basis of many interesting Geo-IR access methods" (Martins et al., 2005).

The STEWARD system (Lieberman et al., 2007) uses and Martins et al. (2005) vote for using separated indexes since they have several advantages, e.g., that queries with either a textual or a geographic part can be efficiently processed, that indexes can be updated separately, and that geographic, thematic, and combined relevance ranking methods are supported (Martins et al., 2005).

To create a hybrid index structure covering spatial and textual information, "any one of [the above mentioned multidimensional indexes] could be used in conjunction with the inverted file structure" (Purves et al., 2007). For instance, Zhou et al. propose "a hybrid index structure, which integrates inverted files and R*-trees, to handle both textual and location aware queries" (Zhou et al., 2005). They compare the use of separate indexes, and two hybrid indexes with an R*-tree for each entry of an inverted index, and an inverted index for each spatial object of the R*-tree, respectively. Their experiments "show that these three structures have almost the same storage cost and [that the hybrid indexes] are superior in query time" (Zhou et al., 2005). Note, however, that they assume that only a single location is associated with each document and that otherwise the storage cost increases rapidly as shown by Purves et al. (2007) in the context of the SPIRIT project. As mentioned above (page 179), they performed similar experiments with more than one region being associated with each document and concluded thus that the two hybrid indexes require much more storage but are faster than separate indexes (Purves et al., 2007).

Finally, Li et al. propose the IR-tree with "its ability to perform [with a top-k document search algorithm] document search, document relevance computation, and document ranking in an integrated fashion" (Li et al., 2011). The IR-tree is based on an R-tree, and the key idea is that the "IR-tree clusters spatially close documents together and carries textual information in its nodes" (Li et al., 2011). Note that the IR-tree

achieves its efficiency because the "strategy is to evaluate the documents based on their joint spatial and textual relevances with respect to a given query q and to terminate the process once the top-k result documents are obtained" (Li et al., 2011). However, several assumptions are necessary. First of all, only a single location can be associated with each document. Thus, if not a single location is to be associated with each document, but all locations mentioned in a document, the IR-tree is hardly applicable. In addition, spatial relevance requires an overlap of a document's location and the query location, and the joint textual and spatial relevance are assumed to be independent and combined as their weighted sum.

Summary

Similar as for temporal information retrieval, geographic information retrieval covers several sub-topics. A popular task is to determine the geographic scope of documents, which led to the fact that several geographic-aware search engines also assume that only a single location is associated with each document. Thus, geographic query constraints are often not compared to all geographic expressions occurring in potentially relevant documents but only to the geographic scopes of the documents. This is a remarkable difference to the approaches in temporal information retrieval.

Note that the focus on single locations associated with documents directly results in a separate calculation of topical and geographic relevance, i.e., to the same weakness as in the approaches to temporal information retrieval. For instance, none of the described approaches takes into account proximity information of relevant terms and relevant location information in the documents' text, a weakness that we will address in our spatio-temporal information retrieval model. In addition, as for temporal information retrieval, there are hardly suitable datasets to evaluate geographic-aware search engines. Although the GeoCLEF and GikiCLEF data sets exist, these are rather challenging due to their question-answering style and the difficulties in determining the respective information needs.

5.2.3 Spatio-temporal Information Retrieval

In contrast to temporal and geographic information retrieval, there is much less work combining both dimensions. In addition, most of the works jointly addressing temporal and geographic information arose quite recently. In the following, we present some spatio-temporal exploration applications, spatio-temporal research competitions, and spatio-temporal search applications.

Spatio-temporal Navigation and Exploration

One of the early works combining temporal and geographic information is GeoTracker (Chen et al., 2007), which reorganizes and aggregates RSS feeds of news documents according to the RSS feed time and locations mentioned in these feeds. For the extraction of geographic information, a rule-based tagger has been developed to extract explicitly mentioned locations. Extracted entities are matched against a location database for normalization purposes. Finally, generic matches are eliminated if a more specific match is extracted in the same feed. Nevertheless, GeoTracker "handle[s] many-to-many relationships [...] between locations and news items" (Chen et al., 2007). In contrast, only a single date is associated with each feed item, namely its creation time. The RSS feeds are visualized on a map and can be traced over time using a time slider. While combining temporal and geographic information, GeoTracker only uses geographic information extraction because the only temporal information is the time of the RSS feed.

Martins et al. (2008) describe an approach to extract and explore temporal and geographic information of textual resources. Note that each item is assigned a single geographic and temporal scope. While they

also apply their approach to RSS news feeds, their focus is on descriptive metadata in digital libraries because the work was developed in the context of the DIGMAP project.[7] In contrast to Chen et al. (2007), they also address the extraction of temporal expressions, although "temporal references [are extracted] at a much simpler level [...] focusing on complete dates and names for historical periods" (Martins et al., 2008). The latter are obviously of special interest in the context of documents about history. Although addressing both, extracted temporal and geographic information, only single geographic and temporal scopes are assigned to each document.

While these works on spatio-temporal exploration detect the geographic and temporal information independently, there are also some works on exploring combined spatio-temporal information. Since these works are more similar to our event-centric search and exploration scenarios than to our spatio-temporal information retrieval model, we survey these works in Chapter 6 and focus now on approaches to spatio-temporal search. For this, we first describe projects to evaluate spatio-temporal search scenarios and then approaches to address the search task.

NTCIR-GeoTime: Geographic and Temporal Information Retrieval Task

NTCIR-GeoTime (Gey et al., 2010, 2011) is an information retrieval research competition which "combines GIR with time-based search to find [documents about] specific events in a multilingual collection" (Gey et al., 2011). The two addressed languages are Japanese and English, and there have been two competitions that took place in 2010 and 2011.

As datasets, newspaper corpora were used covering the years 2002 to 2005 in the first, and the years 1998 to 2005 in the second GeoTime competition. For Japanese, all documents are Mainichi newspaper articles, while the English documents are partially articles of the New York Times corpus (2002 to 2005) and English articles of Xinhua (Chinese), the Korean Times, and Mainichi (1998 to 2001). In total, there are almost 800,000 documents per language.

For both competitions, 25 topics (queries) were created and their development aimed at "creat[ing] topics which were as realistic as possible" (Gey et al., 2010). For this, most topics were generated using Wikipedia pages with listings of notable events for the respective years. Note that although this makes GeoTime "seem to resemble GikiCLEF" (Gey et al., 2010), all topics now contain both, a temporal and a geographic dimension. Nevertheless, they are again formulated in a question answering style and in many topics the temporal and the geographic dimensions are no query constraints but ask for "where" and "when" in a general way. For instance, some topics are of the form *where and when happened X* or *when and where did X die*, while others are rather complex, e.g., *when and where have there been pipeline explosions in an African country with more than 5 fatalities*. We will present further details about the topics in Section 5.6, when evaluating our spatio-temporal information retrieval model.

In addition to the queries, relevance judgments are also available.[8] The top 100 documents of all participants' system have been manually judged by members of the participating groups, resulting in more than 30,000 judgments per language for the 50 queries. Although manual judgments are in four

[7] The DIGMAP project stands for "Discovering our Past World with Digitised Maps" and its "main purpose was to develop a specialized service, reusing metadata from European national libraries, to provide discovery and access to contents provided by those libraries" `http://www.digmap.eu/` [last accessed August 18, 2014].

[8] While the topics of NTCIR-8 and NTCIR-9 GeoTime are available at `http://metadata.berkeley.edu/NTCIR-GeoTime/`, we received the relevance judgments directly from the task organizers.

categories (relevant, temporally relevant, geographically relevant, not relevant), "the three fully and partially relevant categories were aggregated into a single category" (Gey et al., 2010) for the evaluation.

In 2010 and 2011, there have been eight and six teams for Japanese and six and nine teams for English, respectively. In both competitions, a variety of approaches was suggested, e.g., a conventional IR systems "only doing probabilistic ranking coupled with blind relevance feedback, [...] [and a system] which merely counted the number of geographic and temporal expressions found in top-ranked documents of an initial search and then re-ranked based upon initial probability coupled with weighting of the counts" (Gey et al., 2010). In particular in the first round, there were several groups relying on geographic enhancements only. These did not perform as well as those tackling temporal and geographic information. Furthermore, there were typical question answering approaches and while in the first round only few systems made use of external resources, there was a "more extensive use of external resources such as Wikipedia, DBpedia, Geonames, Alexandria Digital Library and Yahoo! PlaceMaker in the second round" (Gey et al., 2011).

Notably, the best-performing team of the second GeoTime manually translated the topics into precise queries – e.g., coordinates of locations were manually checked – to avoid difficulties in query interpretation. However, not only bounding box information for queried regions were manually created, but for some topics "the team constructed queries which included the essence of the answer to the question posed by the topic"(Gey et al., 2011), e.g., by adding place names and specific dates to the queries. This manual translation of the topics is again an indication that one of the main challenges was – similar as in GeoCLEF and GikiCLEF – to correctly understand the question answering style queries. Thus, in the system descriptions of the participating teams, the query interpretation step is often a major part. As mentioned in Section 5.2.2, a typical assumption in the GIR domain is that the geographic aspects of queries can be precisely formulated, e.g., using a map-based interface. Furthermore, we will assume for our approach to address spatio-temporal information retrieval that not only the geographic constraints but also the temporal constraints can be precisely formulated. Thus, the systems of the GeoCLEF, GikiCLEF, and GeoTime participants are only partially comparable to our approach.

MIDR Test Collection – Spatio-temporal IR for Digital Libraries

Palacio et al. (2010) describe an evaluation framework for spatio-temporal information retrieval in the context of digital libraries. For this, they set up the MIDR 2010 Test Collection, a monolingual collection for French. The corpus is relatively small and contains "5,645 paragraphs extracted from 11 books published between the 18th and 20th centuries" (Palacio et al., 2010). In addition, 26 topics (queries) with temporal and geographic constraints and relevance judgments are also made publicly available.[9] To validate their hypothesis that "combining spatial and temporal dimensions along with the topical dimension improves the effectiveness of Information Retrieval systems" (Palacio et al., 2010), they use mono-dimensional systems for each dimension (topic, time, and space). While using any two dimensions outperforms single dimension approaches, the best results are achieved if all dimensions are considered.

Spatio-temporal Search Approaches

The time- and geographic-aware search engines presented in Section 5.2.1 and Section 5.2.2 either address the temporal or the geographic dimension. In the following, we describe IR systems addressing both dimensions.

[9] `http://erig.univ-pau.fr/MIDR/` [last accessed August 19, 2014].

The CITER Project

In the context of the CITER project, Pfoser et al. (2009) developed a search interface to query the content of history textbooks with geographic, temporal, and thematic constraints. For the 55 history textbooks in six languages, the metadata (locations, times, and history categories) is indexed in a relational database. However, while it is pointed out that named entity recognition is applied to "discover temporal, spatial and thematic identifiers that can be linked to a timeline, locations recorded as spatial metadata, and statements referring to concepts in the history ontology" (Pfoser et al., 2009), it remains rather unclear how temporal and geographic expressions are disambiguated and normalized. In addition, no ranking of search results is described and temporal and geographic query constraints are used as boolean filters.

Lucene Extension for Geo-temporal Information Retrieval

A system addressing both dimensions is the Lucene extension LGTE developed and described by Machado et al. (2009).[10] To determine whether a document is relevant to a query with temporal and geographic constraints, it is assumed that "the documents and the queries are assigned to a geographical scope [...] [and that] documents and queries can also be assigned to a temporal interval" (Machado et al., 2009). Thus, the suggested ranking approach considers only a single geographic region and a single time interval associated with each document. For measuring the geographic relevance, "a metric distance [is] computed from the centroid coordinates" (Machado et al., 2009) of a query's and a document's geographic scopes. Similarly, the temporal relevance is determined based on the units of time between the center point of the query time interval and the center point of the document's scope time interval. Finally, to combine the (standard) textual, temporal, and geographic relevance scores, a linear combination is used.

In addition to the temporal and geographic scopes, the extracted temporal and geographic expressions can also be indexed. When calculating the relevance, these are considered as keywords, which, however, "does not improve the results" (Machado et al., 2009). In summary, also information about individual temporal and geographic expressions can be included, the focus lies on using the temporal and geographic scopes of documents. In addition, it is assumed that the textual, temporal and geographic parts of a query are independent of each other, and no proximity between terms satisfying the different query parts is taken into account for determining the relevance of documents for a multidimensional query – i.e., LGTE has a similar weakness as the methods proposed by Berberich et al. (2010) and Purves et al. (2007) for temporal and geographic queries, respectively.

LGTE at GeoTime Competitions

LGTE has also been used in the GeoTime competitions. Cardoso and Silva (2010b) used a "semantic-flavored query reformulation approach" (Cardoso and Silva, 2010a). While all normalized temporal and geographic expressions extracted from the documents are indexed using LGTE, the main idea behind their approach is to detect all kinds of named entities and to exploit "[e]xternal knowledge resources, such as Wikipedia, DBpedia and geographic ontologies [...] to extract information about entities, their properties and relationships among them, and find answers matching the user information need" (Cardoso and Silva, 2010b). This information is then used to enrich the initial queries with the entities that are the answers to the information need. For instance, in a typical GeoTime query such as *where and when did X die*, the system searches for the place and date were X died, e.g., in DBpedia, and then adds the determined place and date entities to the query. Then, the LGTE time and place indexes are searched for these entities.

[10] `http://code.google.com/p/digmap/wiki/LuceneGeoTemporal` [last accessed August 18, 2014].

Machado et al. (2010, 2011) participated in both GeoTime competitions and created textual, temporal, and geographic indexes using LGTE. Hierarchy information about temporal and geographic information is also exploited. For geographic entities, the entity itself and all coarser entities to which it belongs are added to the same index. For instance, the extraction of a city results in adding the city to the geographic index as well as the state and the country in which the city is located. Searching for a country, documents containing only a city located in the country can then also be retrieved. In contrast, "the coverage of hierarchic dates was done at query level using a wildcard" (Machado et al., 2010). For instance, the day expression "May 3, 2010" is indexed as "20100503", and queries for any expressions in 2010 are formulated as "2010*" and match to all entries in the index that start with 2010.

For retrieving and ranking documents, experiments were performed using temporal and geographic expressions either as filters or for query expansion. In addition, all documents without temporal or geographic expressions were removed. Furthermore, separate relevance scores were calculated for each term in each dimension, which resulted in difficulties to combine the relevance scores for a final ranking. The authors thus concluded that "[a]n issue that we must study in the future is the normalization of the scores [...] or more sophisticated techniques for fusion" (Machado et al., 2010). Finally, it is highlighted that "[f]uture work needs to address the topic processing because that could make the real difference" (Machado et al., 2010), as we already summarized for the question answering style research competitions.

Summary

In this section, we surveyed work on spatio-temporal information retrieval. Note that we focused on approaches addressing search and retrieval tasks and that some further works combining geographic and temporal information for exploration tasks will be covered in Chapter 6. In contrast to works on temporal or geographic search and retrieval, there is only little work on combining both dimensions. Furthermore, the approaches developed in the context of the GeoTime competitions are rather tailored to address the task of understanding question answering style queries with temporal and geographic dimensions.

For our approach to spatio-temporal search and retrieval that we develop in this chapter, we assume that both temporal and geographic constraints are explicitly expressed by the user, i.e., that a query consists of a textual, a temporal, and a geographic part. In addition, a key feature of our approach is that we consider the proximity between terms matching the textual, the temporal, and the geographic parts of the query in the retrieved documents. This key feature is not considered by related approaches to temporal, geographic, and spatio-temporal search engines. After developing our multidimensional query functionality in the next section, we will develop the spatio-temporal retrieval model in Section 5.4.

5.3 Multidimensional Querying

Our main goal with respect to multidimensional querying is to add functionality to existing information retrieval applications by allowing the formulation of textual, temporal, and geographic constraints. Usually, when querying large document collections, one is limited to standard text search as known from popular Web search engines like Google, Yahoo!, and Bing. If further features for querying are provided, they are often limited to the metadata of the documents like the date and location of publication. In contrast, we extract directly from the documents' texts temporal and geographic information, and our goal is to allow querying the content of the documents with temporal and geographic constraints. For this, it is important that temporal and geographic constraints can be formulated in a meaningful way.

In this section, we introduce our multidimensional query model consisting of a textual, temporal, and geographic dimension.[11] While we rely on standard textual queries for the textual part of the query, i.e., on existing methods such as Lucene,[12] we formulate some requirements for specifying temporal and geographic queries in Section 5.3.1 and explain our query model for the temporal and geographic dimensions in Section 5.3.2 and Section 5.3.3, respectively.

5.3.1 Requirements

In Section 5.2.2, we mentioned some requirements for geographic querying, e.g., that it should be possible to use a map for specifying a geographic region of interest. In the following, we list some more general requirements for query languages as known in the context of database query languages (Heuer and Scholl, 1991; Saake et al., 2010: p.94–95). These should be considered for both, temporal and geographic queries.

- simple, intuitive, and ad-hoc: First of all, a user should be able to formulate a query intuitively without having to write a complex program (see, e.g., Saake et al., 2010: p.94).
- descriptive: The query language should describe the characteristics of the result set and not how the result set can be created (see, e.g., Saake et al., 2010: p.94).
- safe: Each correct and expressible query should lead to a final result set determined in a final amount of time (see, e.g., Heuer and Scholl, 1991).
- closure: The result of a query has the same form as its input so that queries can be nested (see, e.g., Saake et al., 2010: p.94).
- complete: The query language "should at least have the power of some standard language" (Heuer and Scholl, 1991).

Obviously, in the context of formulating geographic and temporal queries, some of these requirements depend on the provided query interfaces, which will be described in Section 5.3.4 (e.g., simple and descriptive). Other requirements depend on the programmatic realization (e.g., safe and closure), and implementation details how queries are handled will be explained in Section 5.5. In contrast, in Section 5.3.2 and Section 5.3.3, we address the completeness requirement and explain what kinds of temporal and geographic queries should be expressible.

5.3.2 Temporal Querying

The temporal dimension of a query searches the documents' content for specific points in time or time intervals. For the formulation of the temporal dimension of a query, we use the ISO standard also used in the value attribute of TIMEX3 to represent the semantics of an expression. Although one can imagine a formulation of the temporal query in natural language, using the standard format is language-independent and can directly be interpreted so that no temporal tagger has to be used. Thus, a possible error source, i.e., the misinterpretation of the query, can be eliminated. In Section 5.3.4, we will present the query interface used for our prototype. Before that, however, we formally define the temporal query part of our multidimensional query model.

[11] Preliminary studies on temporal and geographic querying have been done in the context of two student projects (Fay, 2011; Fuchs, 2014) and a bachelor thesis (Tobian, 2011).

[12] Lucene, `http://lucene.apache.org/` [last accessed Juli 28, 2014].

As elementary queries, we want to be able to query for time points and time intervals. In addition, combinations of elementary queries shall be expressible, namely the operations of the boolean algebra: conjunction, disjunction, and negation, as well as an arbitrary combination of them. Formally, the temporal query language can be written in Extended Backus-Naur-Form (EBNF) in the following way, with *value* in lines (5.5) and (5.6) describing the standard format of a temporal expression, e.g., "2002-01" for "January 2002", "1999-12-31" for "December 31, 1999", and "2010-H1" for "the first half of 2010".

$$<\text{query}_{temp}> ::= <\text{conjunction}>|<\text{disjunction}>|<\text{negation}>|<\text{point}>|<\text{interval}> \quad (5.1)$$

$$<\text{conjunction}> ::= \underline{(}<\text{query}_{temp}>\underline{\wedge}<\text{query}_{temp}>\underline{)} \quad (5.2)$$

$$<\text{disjunction}> ::= \underline{(}<\text{query}_{temp}>\underline{\vee}<\text{query}_{temp}>\underline{)} \quad (5.3)$$

$$<\text{negation}> ::= \underline{\neg(}<\text{query}_{temp}>\underline{)} \quad (5.4)$$

$$<\text{point}> ::= value \quad (5.5)$$

$$<\text{interval}> ::= \underline{[}(value|\underline{\text{-INF}})\underline{,}(value|\underline{\text{INF}})\underline{]} \quad (5.6)$$

Note that although a <point> is a single expression according to the standard format, it may represent an interval, due to the different granularities of temporal expressions. For example, the value "2002-01" contains all the days, hours, minutes, and seconds of "January 2002". Thus, if <point> is used as the only part of the temporal query, it is handled as an interval with a starting and ending point. For the given example, these are "2002-01-01T00:00:00" and "2002-01-31T23:59:59", respectively. According to this, if a <point> is set as the lower bound or as the upper bound of an interval, the starting point or the ending point of <point> is used, respectively.

5.3.3 Geographic Querying

The geographic dimension of a query is used to search for documents containing references to specific locations. To allow querying regions without a specific name, and for the same reasons as for a temporal query, i.e., to provide language independence of the query and to avoid misinterpretations, we do not use a query formulated in natural language. To formulate the geographic query, we use a map-based approach that allows the user to draw regions on a map, in which she is particularly interested in.

Then, all documents in the document collection are checked for extracted locations whose geographic extent is contained in the specified region. Similar to the temporal query language, one can describe boolean combinations of regions in EBNF, e.g., to exclude a part of a selected region. The regions that are drawn on a map and combined using boolean operators are translated into EBNF, which is formally described below. Although other shapes of regions are possible, we assume that rectangles are drawn on the map. Thus, in lines (5.11) and (5.12), rectangles are used as terminal symbols of the formal language.

$$<\text{query}_{geo}> ::= <\text{conjunction}>|<\text{disjunction}>|<\text{negation}>|<\text{region}> \quad (5.7)$$

$$<\text{conjunction}> ::= \underline{(}<\text{query}_{geo}>\underline{\wedge}<\text{query}_{geo}>\underline{)} \quad (5.8)$$

$$<\text{disjunction}> ::= \underline{(}<\text{query}_{geo}>\underline{\vee}<\text{query}_{geo}>\underline{)} \quad (5.9)$$

$$<\text{negation}> ::= \underline{\neg(}<\text{query}_{geo}>\underline{)} \quad (5.10)$$

$$<\text{region}> ::= rectangle \quad (5.11)$$

$$<\text{region}> ::= rectangle\underline{\setminus}rectangle \quad (5.12)$$

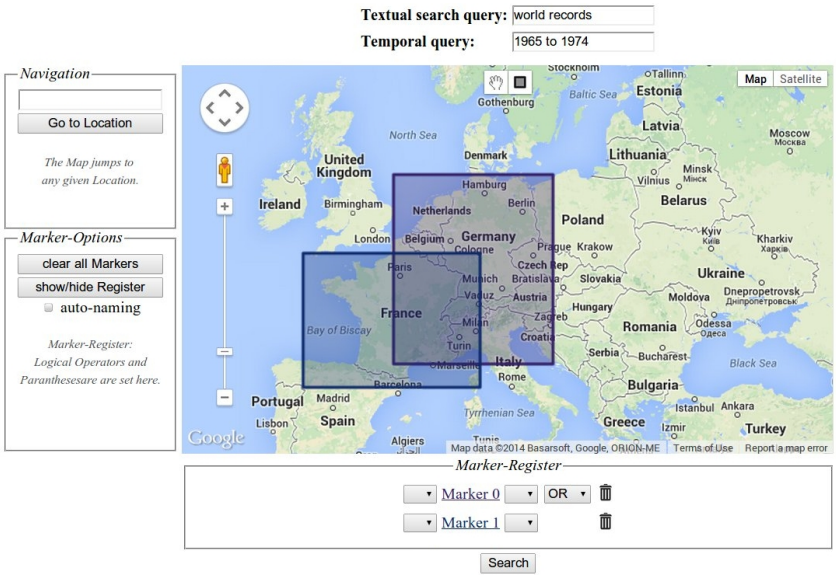


Figure 5.3: Screenshot of the user interface for spatio-temporal search.*

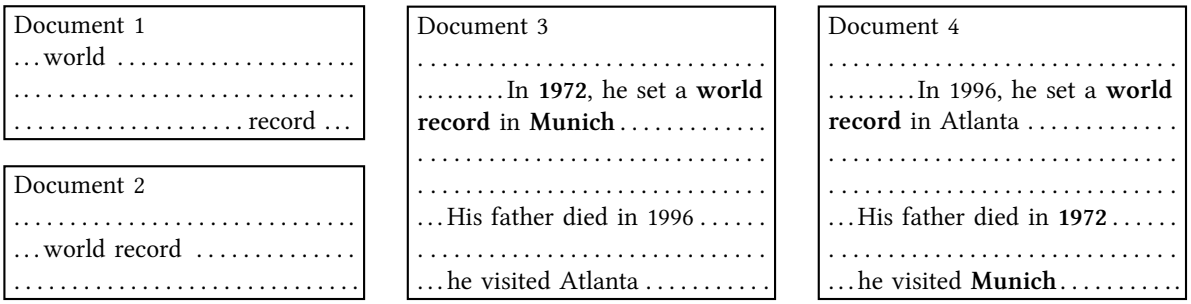
* In contrast to our initial prototypical user interface (Strötgen and Gertz, 2010b), this screenshot is from an updated version and was developed in the context of a student project (Fuchs, 2014). It supports to jump to locations on a map by entering location names for an improved user experience. Furthermore, combining multiple rectangles is now technically supported as it was formally introduced in Section 5.3.3.

In addition to the boolean operators for conjunction ($\wedge$), disjunction ($\vee$), and negation ($\neg$), we define a *without* operator ($\setminus$) to be able to describe regions with holes. Assuming a user wants to select a region R_1 containing a hole R_2, one cannot use the negation operator ("$R_1 \wedge \neg R_2$") since this query would not return documents that contain references to places in R_1 if the document contains one or more references to places in R_2. Compared to the negation operator that defines regions where no locations of a document are allowed, the *without* operator defines a rectangle as a region that is not to be considered. Thus, to query for a region R_1 containing a hole R_2, one uses the query "$R_1 \setminus R_2$".

5.3.4 Query Interfaces

As surveyed in Section 5.2, different types of query interfaces are imaginable for temporal and geographic querying ranging from using natural language expressions to graphical interfaces such as maps and time sliders. To avoid misinterpretations of user information needs, the goal of our query interface is to directly handle normalized temporal and geographic information. In Figure 5.3, the user interface of our prototype for spatio-temporal search is depicted.

For temporal queries, this can be achieved easily since normalized temporal information in the form of TIMEX3 values are quite intuitive. Thus, elementary queries consist of single TIMEX3 values such as "2012" or "1995-05" for querying documents containing references to expressions in the year of 2012 and in May 1995, respectively. Note that although these elementary queries are indeed intervals – assuming the finest granularity of a day, the respective intervals are [2012-01-01,2012-12-31] and [1995-05-01,1995-05-30] – the user does not have to enter the interval boundaries but can use the more intuitive TIMEX3 value



(a) Two example documents. (b) Third example document. (c) Forth example document.

Figure 5.4: Example documents with different term proximities of expressions matching the multi-word query "world record" in (a), and example documents matching the multidimensional information need "world record between 1965 to 1974 in Central Europe" in (b) and (c).

expressions. For the interval boundaries of temporal information needs such as "between 1965 and 1974", the TIMEX3 values can again be used to specify the query as "1965 to 1974".

In contrast to normalized temporal information, normalized geographic information, i.e., latitude and longitude information is not intuitive and users are usually not familiar with specifying them. Thus, neither natural language queries (due to ambiguity issues and due to the fact that not all arbitrary regions can be referred to with a name) nor numerical latitude and longitude queries are optimal choices for geographic query interfaces. To allow for dealing with normalized geographic information without forcing a user to numerically enter this information, our prototype for spatio-temporal search provides a graphical user interface to draw rectangles on a map. For each rectangle, the north-east and south-west latitude and longitude coordinates are determined and can be used for the technical query process.

5.4 Proximity²-aware Ranking Model

As described in detail in Section 5.2, there have been approaches to address temporal and geographic information needs by extracting and normalizing temporal and geographic expressions from documents, and by combining temporal, geographic, and textual queries. These approaches assume that the textual, geographic, and temporal information needs formulated in a query are independent of each other, i.e., independent scores for each part of a query are calculated and finally combined into a single score for ranking documents. However, this independence assumption is problematic because the *proximity among expressions in a document* satisfying the query terms of the different query parts is disregarded.

Generally, the proximity between query terms is a well-known and important feature for textual queries, and probably every standard search engine nowadays considers proximity information in one way or another. In Figure 5.4(a), two example documents matching the multi-word textual query *"world record"* are depicted. Due to the proximity between the two query terms in document 2, this document can be considered as more relevant than document 1. Thus, the query terms of multi-word queries should obviously not be considered as being independent.

Similarly, the independence assumption between expressions matching the different parts of a multi-dimensional query are counter-intuitive and a crucial weakness of existing geographic, temporal, and

spatio-temporal information retrieval approaches. In Figure 5.4(b) and Figure 5.4(c), two example documents are depicted as possibly relevant documents for the multidimensional information need introduced in Section 5.1 (*world record between 1965 and 1974 in Central Europe*). Obviously, document 3 fits the query better than document 4. However, both documents contain exactly the same words as well as the same temporal and geographic expressions. Thus, document 3 can be assigned a higher relevance score than document 4, only if the proximity among expressions in a document satisfying the different query parts is taken into account.

Another aspect hardly considered by related approaches is the *spatial and temporal proximity* of expressions in documents to the temporal and spatial query terms. Assume the simple query *"world record 1990 Germany"*. Also assume there is no document satisfying both, the temporal and spatial information needs, but there are four documents: (A) "In 1991, he set a world record in Germany", (B) "In 1990, he set a world record in France", (C) "In 1991, he set a world record in France", and (D) "In 1992, he set a world record in Japan". Documents (A) and (B) satisfy the geographic and temporal information need, respectively. In addition, they seem to be close to satisfying the information need in general as 1990 is temporally close to 1991, and France is spatially close to Germany. Both documents can thus be considered more relevant than documents (C) and (D). These documents neither satisfy the geographic nor the temporal constraints. However, document (C) is temporally and spatially in closer proximity than "1992" and "Japan" in document (D), so that (C) should be determined as more relevant than (D).

In summary, not only documents (A) and (B) can be ranked in a meaningful way, but also documents (C) and (D) although they neither satisfy the temporal nor the geographic information need. Generally speaking, by considering the semantics of distances and proximities in space and time, the number of ranked documents can be increased because documents not fully satisfying the temporal and geographic queries can be judged based on their distance to the interval or region of interest.

In this section, we will develop a ranking approach that effectively considers both, the proximity of text, temporal, and geographic expressions in documents as well as the spatial and temporal proximity of expressions to query terms. After concisely formulating the problem statement and describing model assumptions and model characteristics in Section 5.4.1, we briefly present a standard ranking for textual queries in Section 5.4.2. Then, the temporal and geographic ranking functions are successively developed. In Section 5.4.7, a measure for incorporating the multidimensional term proximity is defined, before the full model is finally described in Section 5.4.8.

5.4.1 Problem Statement, Model Assumptions, and Model Characteristics

The problem statement for our spatio-temporal information retrieval approach can be formulated as follows:

> **Problem Statement**: Given a document collection D and a search query composed of a textual, a temporal, and a geographic part, return a list of documents $d_i \in D$ ranked by a score measuring how well the combined information need is satisfied. The score should consider the documents' relevance on all parts of the information need and a proximity score covering the distance between terms satisfying the different parts of the information need in the documents. Furthermore, temporal and geographic proximities between temporal and geographic expressions in the query and the documents should be considered to allow a meaningful ranking of documents not fully satisfying all query parts.

Key to our proposed approach is that in a preprocessing step for a given corpus, temporal and geographic expressions in documents are identified as such and normalized in a way that allows for efficient comparison and matching. As indicated in Section 5.1, the main reason for temporal and geographic information not being well handled by standard search engines is that respective expressions in documents are usually treated as regular terms, that is, without any further semantics. To accomplish these preprocessing tasks, a temporal tagger and a geo-tagger are used to extract and normalize temporal and geographic expressions in the documents. While the tasks of geo-taggers and temporal taggers have already been detailed in Section 2.4.3 and Chapter 3, respectively, recall that our contributions to the task of temporal tagging presented in Chapter 3 are crucial for spatio-temporal information retrieval:

- A multilingual temporal tagger is required to process not only English corpora, but also corpora of other languages. Using HeidelTime's current version, temporal expressions can be extracted and normalized in Arabic, Dutch, English, French, German, Italian, Spanish, and Vietnamese corpora.[13]
- A cross-domain temporal tagger is required to successfully process not only news-style documents, but also documents of other domains, e.g., narrative-style documents such as Wikipedia articles (cf. Section 3.3). Without a cross-domain temporal tagger, only explicit temporal expressions could be normalized with high-quality on non-news-style documents, and frequently occurring relative and underspecified expressions would be normalized incorrectly. Using HeidelTime's current version news- and narrative-style documents can be processed with high quality in all supported languages, in addition to English colloquial- and autonomic-style documents.

In contrast to temporal taggers, geo-taggers are not that sensitive to different domains, and there are multilingual geo-taggers available so that we can rely on existing tools for this preprocessing task as will be further detailed in Section 5.6.

Assumption I: Document Profiles

The result of a temporal tagger, when applied to a document, is basically a set of triples, each triple consisting of the term(s) forming the temporal expression t, the offset $p(t)$ of the expression in the document, and the normalized semantics $s(t)$. Similarly, for a geographic expression g found in a document, a geo-tagger returns the expression g, its offset $p(g)$ in the document, and its normalized semantics $s(g)$, which can be a complex object such as a point or a bounding box in addition to some hierarchy information.

For our model, we make the following assumption: Given a document collection D, all documents $d_i \in D$ are preprocessed with a temporal tagger and a geo-tagger. Thus, the temporal and geographic expressions in the documents are extracted and normalized to their standard values (cf. Definition 4.2 and Definition 4.3, page 136). These extracted temporal and geographic expressions are organized in *temporal* and *geographic document profiles* as they have been defined in Section 4.4.1:

- $tdp(d) = \{\langle t_1, s(t)_1, p(t)_1\rangle, ..., \langle t_n, s(t)_n, p(t)_n\rangle\}$
- $gdp(d) = \{\langle g_1, s(g)_1, p(g)_1\rangle, ..., \langle g_m, s(g)_m, p(g)_m\rangle\}$

In Section 5.5, we will give more details on how these profiles are computed and managed for a given document collection in a preprocessing step for subsequent efficient lookup and ranking tasks.

[13] Some further languages have already been added to HeidelTime and other languages are under development (cf. Section 3.7).

Assumption II: Queries

A further assumption for our model concerns the queries. We assume that a query consists of a textual part q_{text} (terms), a temporal part q_{temp} (one or more time intervals), and a geographic part q_{geo} (one or more geographic regions specified by, e.g., bounding boxes). Thus, we define a query as:

- $q = \{q_{text}, q_{temp}, q_{geo}\}$

It should be noted that the user can specify such a query in different ways, depending on what query interface is provided. As described in Section 5.3.4, for a normal textual query, geographic and temporal expression (including time intervals such as "1999 to 2011") are identified and normalized, very much in the same way as expressions in documents are handled. One can also envision a graphical query interface in which the user specifies a point location or a bounding box plus some time interval using a time-slider. Here, one would already obtain normalized values for respective query components.

The document profiles are used to evaluate q_{temp} and q_{geo} and to determine the temporal and geographic proximities – based on normalized values – between expressions in the documents and the query parts. Again, implementation details on how to efficiently process q_{temp} and q_{geo} will be given in Section 5.5.

Model Characteristics

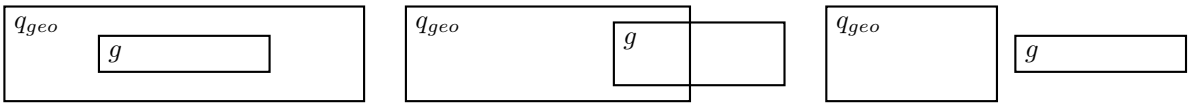
Based on these assumptions, we incrementally develop in the next sections the proximity^2-aware ranking model for textual, temporal, and geographic information needs formulated in search queries. Before that, however, we briefly summarize the key characteristics of the model:

- For the individual components q_{text}, q_{temp}, and q_{geo} present in a search query, single scores are calculated.
- Given a document, based on the distances between terms and expressions in the document satisfying the different query parts, a score is calculated (*term proximity score*).
- There will typically be documents not directly satisfying the q_{temp} and q_{geo} parts of a query. For such documents, still positive temporal and geographic scores can be calculated. This is because the temporal and geographic distances between expressions in the documents and the time interval and region specified in a query are taken into account (*temporal and geographic proximity*).

5.4.2 Textual Ranking

One part of our proximity^2-aware ranking model is to calculate a score s_{text} for the textual part q_{text} of a query. For this, we use Okapi BM25 (Robertson et al., 1994), a well-known standard measure for ranking documents according to a textual query. This measure is mainly based on the term frequency $c(w,d)$ and a normalized version of the inverse document frequency (first fraction in Equation 5.13, with $df(w)$ being the number of documents containing term w). For the text part q_{text} of a query and a document $d \in D$ with $|D| = N$, it is defined as follows:

$$s_{text}(q_{text}, d) := \sum_{w \in q_{text} \cap d} ln \frac{N - df(w) + 0.5}{df(w) + 0.5} \times \frac{(k_1 + 1) \times c(w,d)}{k_1((1-b) + b\frac{d_{len}}{D_{avgl}}) + c(w,d)} \tag{5.13}$$



(a) Expression g inside q_{geo}. (b) Expression g overlaps q_{geo}. (c) Expression g outside q_{geo}.

Figure 5.5: Possible relations between a geographic expression and a geographic query (q_{geo}).

Note that the score is length-normalized using the length d_{len} of a document d and the average document length D_{avgl} of all documents in D. The parameters $k_1 \in [1.2, 2.0]$ and $b = 0.75$ calibrate term frequency scaling and length normalization scaling (Manning et al., 2008: p.215). For every document $d_i \in D$, this formula determines a textual score $s_{text}(q_{text}, d_i)$ representing the relevance of document d_i with respect to q_{text}. Based on these key concepts of this ranking formula, in the following, we develop our ranking functions for q_{temp} and q_{geo}.

5.4.3 Temporal and Geographic Ranking

Similar to the score s_{text}, we want to calculate the scores s_{temp} and s_{geo} representing how well a document satisfies the other two query parts q_{temp} and q_{geo}, respectively.

A difference between validating q_{temp} and q_{geo} and validating q_{text} is that q_{temp} and q_{geo} may consist of one or more time intervals/geographic regions and q_{text} consists of one or more terms. More importantly, the regular terms considered in s_{text} and intervals/regions have different characteristics. While the terms matching q_{text} directly occur "as is" in documents (after preprocessing such as stemming), expressions that may match q_{temp} and q_{geo} have to be validated based on their normalized semantics and, in addition, do not necessarily have to completely match q_{temp} and q_{geo}. These differences have to be taken into account when calculating s_{temp} and s_{geo} following the idea of computing s_{text}.

Given q_{temp} and q_{geo} and the document profiles $tdp(d)$ and $gdp(d)$ of a document d, the normalized semantics of expressions in $tdp(d)$ and $gdp(d)$ can (i) be inside q_{temp}/q_{geo}, (ii) overlap q_{temp}/q_{geo}, or (iii) be outside q_{temp}/q_{geo}. Assume, for example, the normalized query time interval $q_{temp} = [1965,1974]$. The expression "1972" corresponds to case (i), "1960s" to case (ii), and "1960" to case (iii). Note that for case (i), an expression may cover different parts of a query interval/region. For example, "September 1972" and "1972" are both in q_{temp} but cover different parts of it due to their different granularities. Accordingly, Figure 5.5 contains visualizations of the three possible relations between a geographic expression g and q_{geo}, depicted as rectangles.

Berberich et al. (2010) assume for their approach to satisfy temporal information needs (cf. Section 5.2.1) that the more of the query time interval is covered by a temporal expression, the more relevant the temporal expression (Berberich et al., 2010). However, we argue that it is only important whether an expression is within q_{temp}/q_{geo} or not, and that the coverage of q_{temp} and q_{geo} can be better determined based on *all* expressions in $tdp(d)$/$gdp(d)$, as we will justify in Section 5.4.5.

5.4.4 Temporal and Geographic Proximity

Due to the three different ways of how expressions in a temporal document profile $tdp(d)$ and a geographic document profile $gdp(d)$ are related to a query part q_{temp} and g_{geo}, respectively, we cannot simply use the

term frequency as for regular terms matching q_{text}. Thus, we calculate the *weighted value frequencies*, vf_t and vf_g, which aggregates the *value weight* vw_t/vw_g of every expression $t \in tdp(d)$ and $g \in gdp(d)$. The weighted value frequency for temporal expressions is shown in Equation 5.14:

$$vf_t(q_{temp}, d) := \sum_{v(t) \in tdp(d)} vw_t(v(t), q_{temp}), \text{ with} \tag{5.14}$$

$$vw_t(v(t), q_{temp}) := \begin{cases} 1, & \text{if } v(t) \text{ is in } q_{temp} \quad \text{(i)} \\ \frac{|v(t) \cap q_{temp}|}{|v(t)|}, & \text{if } v(t) \text{ overlaps } q_{temp} \quad \text{(ii)} \\ exp^{-\frac{\delta(v(t), q_{temp})}{|q_{temp}|}}, & \text{if } v(t) \text{ is outside } q_{temp} \quad \text{(iii)} \end{cases}$$

Accordingly, the weighted value frequency for geographic expressions is shown in Equation 5.15:

$$vf_g(q_{geo}, d) := \sum_{v(g) \in gdp(d)} vw_g(v(g), q_{geo}), \text{ with} \tag{5.15}$$

$$vw_g(v(g), q_{geo}) := \begin{cases} 1, & \text{if } v(g) \text{ is in } q_{geo} \quad \text{(i)} \\ \frac{|v(g) \cap q_{geo}|}{|v(g)|}, & \text{if } v(g) \text{ overlaps } q_{geo} \quad \text{(ii)} \\ exp^{-\frac{\delta(v(g), q_{geo})}{|q_{geo}|}}, & \text{if } v(g) \text{ is outside } q_{geo} \quad \text{(iii)} \end{cases}$$

The first two cases (i) and (ii) are straightforward: if $v(t)/v(g)$ is inside q_{temp}/q_{geo}, we want vw_t/vw_g to be 1. If $v(t)$ or $v(g)$ overlaps q_{temp}/q_{geo}, we want vw_t/vw_g to represent the proportion of $v(t)$ or $v(g)$ being inside of q_{temp} or q_{geo}. For example, given $q_{temp} = [1965,1974]$, then $vw_t(\text{“1960s”}) = 0.5$ and $vw_t(\text{“20th century”}) = 0.1$.

For the third case (iii), however, that is, if $v(t)$ is outside q_{temp} or $v(g)$ is outside of q_{geo}, we do not want vw_t/vw_g to be simply 0 as we want to distinguish whether or not $v(t)/v(g)$ are temporally/spatially close to q_{temp}/q_{geo} or not. Thus, we introduce the first important proximity parameter of our model to calculate the distance δ between the normalized value $v(t)/v(g)$ and q_{temp}/q_{geo}. This allows to score also documents with a $s_{temp} > 0$ that do not contain expressions directly satisfying q_{temp} and with $s_{geo} > 0$ that do not contain expressions directly satisfying q_{geo}.

We use the distance δ in relation to the “size” of q_{temp} and q_{geo}, which is denoted $|q_{temp}|$ and $|q_{geo}|$, respectively. In the case of temporal expressions, then, depending on the granularity of q_{temp}, it is the number of days, months, years, etc. covered by q_{temp}. Given two temporal queries q_{temp-1} and q_{temp-2} of the same granularity, intuitively, the following condition should hold:

if $\delta(v(t), q_{temp-1}) = \delta(v(t), q_{temp-2})$ **and** $|q_{temp-1}| < |q_{temp-2}|$
then $vw_t(\delta(v(t), q_{temp-1})) < vw_t(\delta(v(t), q_{temp-2})$

In other words, the same distance between a normalized value $v(t)$ and a normalized temporal query should result in a lower value weight if the size of the query interval is smaller. For example, assume $v(t) =$ 1972-09-03 and two temporal query parts $q_{temp-1} = [$1972-08-01, 1972-08-31$]$ and $q_{temp-2} =$

[1972-08-30, 1972-08-31]. The distance to $v(t)$ is the same for both queries (3 days). However, due to the larger interval of interest formulated by q_{temp-1} (31 days), the distance of 3 days is less relevant than in the second case, where the size of q_{temp-2} is smaller (2 days).

Similarly, in the case of geographic expressions, the size of q_{geo} is simply the area described by q_{geo} (based on its normalized semantics), and given two geographic queries q_{geo-1} and q_{geo-2}, intuitively, the following condition should hold:

if $\delta(v(g), q_{geo-1}) = \delta(v(g), q_{geo-2})$ **and** $|q_{geo-1}| < |q_{geo-2}|$
then $vw_g(\delta(v(g), q_{geo-1})) < vw_g(\delta(v(g), q_{geo-2})$

Again, as in the case of temporal proximities, the same distance between a normalized value $v(g)$ and a normalized geographic query should result in a lower value weight if the size of the query region is smaller. Examples for normalized geographic expressions found in documents and two query regions are devised similarly as for temporal expressions in our framework, based on the shortest distance between respective regions and the area of regions. For geographic expressions, it obviously becomes even simpler in case only geographic points (as normalized values) are considered.

The desired behavior of the value weight functions $vw_t(v(t), q_{temp})$ and $vw_t(v(g), q_{geo})$ with $\delta > 0$ can be described as follows: the smaller $\frac{\delta(v(t),q_{temp})}{|q_{temp}|}$ and $\frac{\delta(v(g),q_{geo})}{|q_{geo}|}$, the lower $vw_t(v(t), q_{temp})$ and $vw_g(v(g), q_{geo})$, respectively, with its first derivatives being negative and its second derivatives being positive. This concave behavior is obtained by exponential terms of the form $exp(-\frac{\delta(v(t),q_{temp})}{|q_{temp}|})$ and $exp(-\frac{\delta(v(g),q_{geo})}{|q_{geo}|})$, so that we can summarize the behavior of the value weight functions $vw_t(v(t), q_{temp})$ and $vw_g(v(g), q_{geo})$ as defined in Equation 5.14 and Equation 5.15.

In summary, an important ingredient of our novel ranking model is that for the temporal and geographic ranking functions, we use the weighted value frequency instead of the standard term frequency for regular terms. This approach appropriately considers the semantics of temporal and geographic expressions in terms of proximity of time intervals and geographic regions, respectively, based on well-defined distance metrics for time and space.

5.4.5 Coverage of the Query Interval and Region

In the Okapi BM25 for the textual ranking score s_{text}, the second important feature besides the term frequency is the inverse document frequency. It carries information about how characteristic a term is for a document with respect to the document collection.

For our modifications to BM25 for calculating the temporal and geographic scores s_{temp} and s_{geo}, we combine information about the document collection with information about the coverage of the query interval/region. Given q_{temp}/q_{geo} and a document profile $tdp(d)$/$gdp(d)$, we calculate the ratio of distinct normalized values in the document profile and the number of distinct normalized values in the combined document profile of all documents $tdp(D)$/$gdp(D)$ overlapping q_{temp}/q_{geo}.

To avoid that the coverage is zero or undefined if a document or the document collection contains no normalized values overlapping with q_{temp}/q_{geo}, we add 0.5 to both counts. This is important since temporal and geographic scores should be positive in both cases for the temporal and geographic proximity

introduced above to work effectively. By this, the coverage of a document without values overlapping with q_{temp}/q_{geo} is larger than 1 and the coverage is the same for all documents if no values in the document collection overlap with q_{temp}/q_{geo}. Formulas for calculating the temporal coverage and the geographic coverage are shown in Equation 5.16 and Equation 5.17, respectively.

$$coverage_t(d, q_{temp}) := \frac{count_{dist}(v(t) \in tdp(d) : v(t) \cap q_{temp} \neq \emptyset) + 0.5}{count_{dist}(v(t) \in tdp(D) : v(t) \cap q_{temp} \neq \emptyset) + 0.5} \tag{5.16}$$

$$coverage_g(d, q_{geo}) := \frac{count_{dist}(v(g) \in gdp(d) : v(g) \cap q_{geo} \neq \emptyset) + 0.5}{count_{dist}(v(g) \in gdp(D) : v(g) \cap q_{geo} \neq \emptyset) + 0.5} \tag{5.17}$$

For example, given a temporal query "August 1972" and two documents with the first containing some temporal expressions referring to "1972-08-01" and the second containing some expressions referring to "1972-08-07" and "1972-08", respectively. In addition, in the corpus, there are ten distinct normalized values of temporal expressions that (partially) match the temporal query. Then, the temporal coverage of the first document is $\frac{1.5}{10.5}$ and the temporal coverage of the second document is $\frac{2.5}{10.5}$.

In our opinion, when being faced with a temporal or geographic query formulated as time interval or geographic region, the most relevant document does not necessarily cover the whole interval or region but contains many different normalized values in the interval or region of interest compared to other documents. Thus, we use Equation 5.16 and Equation 5.17 as corpus-dependent coverage instead of using the plain coverage of q_{temp} and q_{geo} or the inverse document frequency as for terms.

5.4.6 Temporal and Geographic Ranking Scores

Replacing the inverse document frequency by the temporal/geographic coverage and the term frequency c by the weighted value frequencies vf_t and vf_g in Equation 5.13, the temporal and geographic scores s_{temp} and s_{geo} are now calculated as shown in Equation 5.18 and Equation 5.19. In the same way as s_{text} is defined in Equation 5.13, these scores are length-normalized, and the parameters k_1 and b are used to calibrate the scaling behavior.

$$s_{temp}(q_{temp}, d) := \sum_{v \in q_{temp}} coverage_t(d, v) \times \frac{(k_1 + 1) \times vf_t(v, d)}{k_1((1 - b) + b\frac{d_{len}}{D_{avgl}}) + vf(v, d)} \tag{5.18}$$

$$s_{geo}(q_{geo}, d) := \sum_{v \in q_{geo}} coverage_g(d, v) \times \frac{(k_1 + 1) \times vf_g(v, d)}{k_1((1 - b) + b\frac{d_{len}}{D_{avgl}}) + vf(v, d)} \tag{5.19}$$

Thus far, we have defined scores to rank the individual components q_{text}, q_{temp} and q_{geo}, where for ranking q_{temp} and q_{geo} we introduced temporal and geographic proximity measures based on the distances of time intervals and geographic regions, respectively. We now turn to the second type of proximity measure, the term proximity, as another important ingredient to our ranking model.

5.4.7 Multidimensional Term Proximity

The relevance scores described in the previous sections represent independent scores for the query parts q_{text}, q_{temp}, and q_{geo} with respect to a document. While previous approaches combine such independent scores into a final ranking score for a document, we argue that this independence assumption is problematic. As illustrated in the examples in Figure 5.4 (page 193), information about the proximity in a document between terms and expressions satisfying q_{text}, q_{temp}, and q_{geo} should be considered to reward documents in which the proximity among matching expressions is small, and to penalize documents where such a proximity is large.

For multi-word textual queries, using proximity information is a widely used feature to improve retrieval models (see, e.g., Rasolofo and Savoy, 2003). Tao and Zhai analyzed different ways to measure the proximity between terms matching a textual query in documents (Tao and Zhai, 2007). In their comparison of five measures, the minimum pair distance (shortest distance of two different query terms, independent of the number of query terms) performed best. Although we are faced with a slightly different problem here, because the terms and expressions for which we want to measure the proximity are of different types, we use their study as basis for developing the function to calculate the proximity score s_{prox}.

For this, we transfer the minimum pair distance into a *minimum triple distance*. Given a document d, such a distance then is naturally defined as the shortest distance among a term w of q_{text}, a temporal expression t satisfying q_{temp}, and a geographic expression g satisfying q_{geo}, denoted $prox(w, t, g, d)$. Clearly, the closer w, t, and g are together in document d, the higher should be the ranking for that document with respect to the query.

In contrast to the original proximity measure for two terms, there is no need that the three terms or expressions w, t, and g, respectively, occur within a few tokens, but it should be awarded if they occur within a few sentences. Thus, instead of the original concave function, we use the following proximity transformation function (containing cubic terms in both nominator and denominator):

$$s_{prox}(q, d) := exp(\frac{ln(0.5) \times prox(w, t, g, d)^3}{50^3}) \tag{5.20}$$

The behavior of Equation 5.20 is shown in Figure 5.6. Assuming a typical sentence length of 20 to 25 tokens (Manning and Schütze, 2003: p.136), the function only slightly penalizes proximities within one or two sentences, but significantly penalizes proximities larger than three sentences since s_{prox} is convex for proximities smaller than 50 tokens and concave for larger proximities.

5.4.8 Full Multidimensional Proximity²-aware Ranking Model

Having defined the separate scores for the textual, temporal, geographic, and proximity ranking, we are now finally faced with the same problem as similar approaches not considering proximity information, namely, how to combine the single scores in a meaningful way. A typical way that also allows to specify weightings for the single scores is to use a linear combination. For this, we first normalize the s_{text}, s_{temp}, and s_{geo} scores by the maximum score for the given query, denoted $\hat{s}_{text}(q)$ etc. Thus, for each query the highest textual, temporal, and geographic scores are set to 1. The proximity score is already normalized, as described in the previous section. We therefore obtain the following score for a query q and document d:

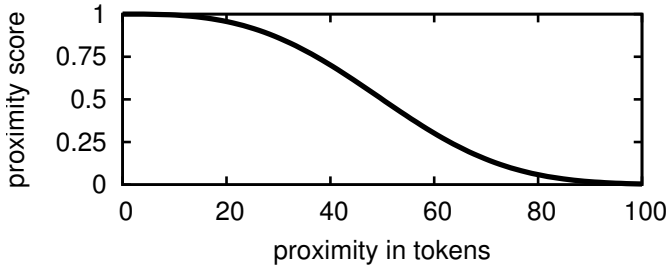


Figure 5.6: Ideal shape of the proximity transformation function.

$$s(q,d) := (1-\alpha_t-\alpha_g)\frac{s_{text}(q,d)}{\hat{s}_{text}(q)} + \alpha_t\frac{s_{temp}(q,d)}{\hat{s}_{temp}(q)} + \alpha_g\frac{s_{geo}(q,d)}{\hat{s}_{geo}(q)} + \beta s_{prox}(q,d) \tag{5.21}$$

The parameters α_t and α_g are used to weight the three query components q_{text}, q_{temp}, and q_{geo}. In addition, β is used to weight the proximity measure. In our evaluation, which is detailed in Section 5.6, we show the impact of varying β and analyze the influence of the proximity feature of our model. Before that, we present some extraction, indexing, and querying details in the next section.

5.5 Extraction, Indexing, and Querying Details

In Section 5.5.1, we briefly describe how temporal and geographic expressions are extracted from text documents and normalized to some standard format. After that step, temporal, geographic, and event document profiles can be stored and indexed as will be explained in Section 5.5.2. Finally, we present in Section 5.5.3 how the document profiles and further indexing strategies are used to process queries with our proximity^2-aware ranking model.

5.5.1 Extraction and Normalization of Geographic and Temporal Information

The procedure to extract geographic and temporal expressions from text documents for our spatio-temporal information retrieval approach is identical as the procedure to extract spatio-temporal events which was explained in Section 4.5.1 – except that extracted temporal and geographic information is combined when extracting spatio-temporal events from documents.

Thus, we process the documents of the document collection as depicted in Figure 4.4 (page 151) without the cooccurrence extractor. That is, each document is processed by Yahoo! Placemaker and HeidelTime to extract and normalize geographic and temporal expressions, respectively. In addition, we implemented a CAS Consumer that directly stores the extracted information in a database so that the indexes described below can be easily created.

5.5.2 Storing Temporal, Geographic, and Event Document Profiles

In Section 4.4, we introduced the concepts of temporal, geographic, and event document profiles. Now, we explain how these profiles are stored to efficiently access them.

For storing the profiles, we use a Postgres database and index structures provided by PostGIS.[14] The stored information is separated into temporal and geographic entities and textual references to these entities. While it is often not important to analyze the words occurring in the documents, normalized information is of major interest. Thus, the database, in which the CAS consumer of our document processing pipeline inserts extracted information, contains the following tables for temporal and geographic information as well as for events – if these are extracted additionally: temporal entities, geographic entities, and event entities as well as temporal references, geographic references, and event references.

In the entities tables, all information about the semantics of the spatio-temporal events, the points in time, and the places are stored, i.e., normalized value information, normalized containment information, and latitude/longitude information. The three reference tables contain pointers to entries of the entities tables as well as all the information about the expressions such as document ids, character offsets, confidence values, and all further types of information that is expression-dependent. For instance, assuming the temporal expression *"October 14, 2014"* is extracted from document d at offset i to j. Then, an entry for the normalized temporal information is added to the *temporal entities* table (2014-10-14) – if it does not already exist – and an entry to the *temporal references* table is added and contains a reference to the "2014-10-14" entry in the temporal entities table in addition, to d, i, and j.

To access a temporal, geographic, or event document profile of a specific document, one can simply select all entries in the respective references tables with a specific document id. In addition, temporal, geographic, and event information can be easily queried with temporal and geographic constraints. To guarantee that querying is efficient, in addition to the document ids, the normalized temporal and geographic information has to be indexed. For this, we use PostGIS' built-in indexes – variations of the B-tree and R-tree index structures (see, e.g., Rigaux et al., 2002) for temporal and geographic information, respectively.[15] As was shown in related works in geographic information retrieval, using an R-tree (variation) is a good choice for handling geospatial data efficiently (see, e.g., Martins et al., 2005). As an R-tree for multidimensional data, B-tree variations are efficient indexes for one-dimensional data.

5.5.3 Indexing and Querying Strategies

In this section, we explain the indexes and querying strategies for our spatio-temporal information retrieval approach.

Indexing Textual Content

For our proximity^2-aware ranking model, not only temporal and geographic document profiles are required but also efficient index structures for textual content. Thus, in addition to processing the documents of the document collection with the temporal tagger and geo-tagger, we also apply the porter stemmer[16] to the documents and remove stop words. For the stemmed words, a standard inverted index with term frequency information is used (see, e.g., Manning et al., 2008: p.5). Additionally, document frequency and document length information are indexed. This is already sufficient to calculate s_{text} as shown in Equation 5.13. To calculate also the term proximity feature of our proximity^2-aware ranking model, we add the position information to each term/document pair.

[14] PostGIS (version 1.5), `http://postgis.net/docs/manual-1.5/index.html` [last accessed November 15, 2014].

[15] Instead of an R-tree, we use a GiST (Generalized Search Tree) index, a built-in R-tree variation, for indexing geographic data; `http://postgis.net/docs/manual-1.5/ch04.html#id361810` [last accessed November 15, 2014].

[16] `http://tartarus.org/~martin/PorterStemmer/` [last accessed November 15, 2014].

Indexing Geographic and Temporal Information for Spatio-temporal Search

Although one could use just the PostGIS index structures as presented above, we run the following indexing strategy. As related approaches suggested, it is useful to combine these indexes with inverted indexes (see, e.g., Purves et al., 2007). To validate geographic queries, the GiST index is used to determine all geographic entities that satisfy q_{geo}. Then, an inverted index returns all documents that contain expressions referring to these geographic entities including frequency information.

For references to temporal entities, we create multiple inverted indexes – namely year-, month, and day-level inverted indexes with value frequency information. Following this strategy, we exploit the temporal hierarchy information which makes the querying much more efficient. Note that all fine-grained values are additionally included in the indexes for coarser granularities, e.g., an expression normalized to "1972-08-01" is listed in the day-level index, and as "1972-08" and "1972" in the month- and year-level indexes. Depending on the granularity of the query, the respective index is searched to return all relevant documents. Assuming a temporal query is specified as "1994 to 1996", then only three entries of the year-level index have to be returned. In contrast when querying all temporal entities, up to 1,134 temporal entities are determined as relevant (three year values, 36 month values, and 1,095 day values). Thus, three postings lists have to be handled instead of up to 1,134. Note that each entry in each index does not only contain a reference to the documents but also frequency information. In addition, for calculating the term proximities, we also index the position information for each value/document pair.

Multidimensional Query Processing

As described in Section 5.3, we assume that the query to use our proximity^2-aware ranking model contains a textual part (q_{text}), a temporal part (q_{temp}), and a geographic part (q_{geo}).

The textual query contains regular words and is processed in the same way as the documents, i.e., using the Porter stemmer and removing stop words. For q_{temp}, we assume that the intervals are specified using normalized values, e.g., *"2001-11 to 2001-12"*. Thus, the normalized values can directly be used to determine relevant temporal entities using the index of respective granularity. For all map-based geographic queries that can be formulated according to our query model, the normalized latitude/longitude information of the rectangles can also be directly used to determine all geographic entities that are within the query region.

Calculating Relevance Scores

To calculate the final relevance scores for retrieved documents, we first calculate s_{text}. The scores s_{geo}, s_{temp}, and s_{prox} are only computed for the top-k documents of the text query. For our evaluation presented in the next section, we set k to 2000 and perform a re-ranking of the top-2000 ranked documents.

The temporal and geographic relevance scores are calculated based on the indexing and querying functionality detailed above. However, the weighted value frequencies for expressions not satisfying q_{temp} or q_{geo} (Equation 5.14 and Equation 5.15) is only calculated for those documents that do not have any normalized values of temporal/geographic expressions directly satisfying q_{temp} or q_{geo}. This makes the calculation of s_{temp} and s_{geo} much more efficient. To calculate the weighted value frequencies, we directly access the temporal and geographic document profiles of the documents using the database indexes and iterate over all temporal and geographic expressions of the respective documents.

Using the indexes and strategies described in this section, we also performed the evaluation of our proximity^2-aware ranking model as will be detailed in the next section.

5.6 Evaluation

Evaluating a combination of textual, temporal, and geographic information needs is difficult. First, there are no benchmarks from IR challenges such as TREC[17] in which a query consists of a textual, a temporal, and a geographic part. Second, setting up a new evaluation scenario is time- and labor-intensive in particular due to the difficulty of collecting relevance judgments. Note that it would not be sufficient to judge the top-k ranked documents of the final model for each query, but relevance judgments would have to be collected additionally for the top-k ranked documents of baselines and model variations so that the full model can be compared and evaluated in a meaningful way.

In the context of geographic information retrieval, it was pointed out "that evaluating geographic relevance is more difficult than thematic relevance" (Purves et al., 2007) since it is often difficult to determine geographic relevance without knowledge of the area described by the information need. Furthermore, even when considering thematic and geographic relevance without temporal relevance, "test collections can only be built by large cooperative projects" (Purves et al., 2007). Thus, in the context of this thesis, setting up a new evaluation scenario for textual, temporal, and geographic information needs is not feasible so that we make use of an existing benchmark although it is not directly tailored to our task of spatio-temporal information retrieval so that some modifications are necessary.

The probably most related data sets covering temporal and geographic aspects are the data sets of the NTCIR GeoTime challenges (Gey et al., 2010, 2011) so that we use GeoTime data for our evaluation. As described in Section 5.2.3, a major difficulty in the GeoTime competitions was to correctly interpret the queries in general, as well as the temporal and geographic aspects of the queries. Each query either contains explicit temporal and geographic expressions (e.g., *"2002"* and *"China"*) or represents the temporal and geographic information needs in the form of asking for *"where"* and *"when"*. Since our model expects that temporal and geographic constraints are concisely formulated, some adaptations to the textually formulated queries of the GeoTime data are necessary as well as some minor adaptations to our model.

Due to the required adaptations, we do not aim at performing an evaluation that shows that our model outperforms other spatio-temporal information retrieval models, e.g., the systems of the GeoTime participants. In contrast, we aim at an evaluation that demonstrates the usefulness of the features of our model. In particular, we want to validate the following hypotheses:

- Considering normalized temporal and geographic information extracted from the content of the documents improves spatio-temporal information retrieval.
- Combining topical, temporal, and geographic relevance scores outperforms a topical ranking combined with boolean temporal and geographic filtering.
- The term proximity feature of our model improves the evaluation results.

[17] `http://trec.nist.gov/` [last accessed October 1, 2014].

The third hypothesis is particularly interesting because the term proximity feature is a major distinction between our model and related models that all ignore the dependencies between query components. We evaluate this feature by determining the influence of using different weights of the textual proximity score of our model. To analyze the first two hypotheses, we compare our proximity^2-aware ranking model with two baselines, which will be described in Section 5.6.1. Then, we describe the GeoTime data and required modifications in Section 5.6.2 followed be an explanation of our model adaptations (Section 5.6.3). Finally, we detail the evaluation results in Section 5.6.4.

5.6.1 Baselines

In our experiments, we compare the evaluation results of our proximity^2-aware ranking model with the following two baselines:

- BL-text: As first baseline, we handle temporal and geographic information needs in the most simple way. Explicit temporal and geographic expressions are included in the textual part of the queries without treating the temporal and geographic expressions in a special way – neither in the queries nor in the documents.
- BL-bool: As second baseline, temporal and geographic expressions are extracted and normalized, and the temporal and geographic information needs are formally described as time intervals and regions, respectively. Then, documents are ranked according to q_{text} while q_{temp} and q_{geo} are used as boolean constraints. Thus, all documents not satisfying the temporal and geographic information needs are discarded from the results. If there are no explicit temporal or geographic constraints, documents without any temporal or geographic expressions are discarded.

The first baseline can be considered as a standard text search engine. In contrast, the second baseline is a very strong baseline, which uses the semantics of temporal and geographic expressions – a feature usually not used by standard search engines. In the next section, we will present, among others, the original textual GeoTime queries as well as the modified queries used for the two baselines and for the proximity^2-aware ranking model.

5.6.2 GeoTime Data and Modifications

For the 25 queries of the NTCIR-8 GeoTime dataset, there are 17,423 judgments in total.[18] Many of the queries are of the form *"where and when happened X"*, but there are also some queries with explicit temporal constraints, geographic constraints, or both. Table 5.1 gives an overview of the types of explicit constraints as well as the number of positively judged documents for each topic.

Note that the varying numbers of positive judgments directly indicate the different levels of difficulty since judgments exist for the top 100 ranked documents of each system of the GeoTime participants. In addition, note that we only use these judgments in our experiments and consider retrieved documents without any judgment as not relevant. This guarantees that no biased judgments are included and that our evaluation results are calculated on publicly available data only.

As mentioned above, the queries of the NTCIR-8 GeoTime data set are formulated as questions. In contrast, our focus is not to correctly interpret question-style queries, and our proximity^2-aware ranking

[18] More general information about the GeoTime data as well as references and links were provided in Section 5.2.3.

topic	explicit constraints	positive judgments	topic	explicit constraints	positive judgments	topic	explicit constraints	positive judgments
01		9	10		10	19		79
02	geo	335	11	time	96	20		9
03		5	12		36	21	time	3
04		38	13	geo	18	22	time	15
05		8	14	geo, time	31	23	geo	27
06	geo	112	15	time	71	24		48
07		8	16		320	25	geo	19
08	geo	172	17	geo	24			
09		49	18	time	58			

Table 5.1: NTCIR-8 GeoTime topics with explicit constraints and the number of positive judgments.

model assumes that an information need is described as a multidimensional query with a textual part, a temporal part, and a geographic part. Thus, for our experiments, we reformulated the queries. In Table 5.2, the original question-style queries and the queries used to evaluate the two baselines and the proximity^2-aware ranking model are listed. For the experiments with the proximity^2-aware ranking model and for the baseline BL-bool experiments, explicit geographic constraints are provided as bounding box information for the described regions.[19] This corresponds to how geographic queries are specified in our multidimensional query model.

5.6.3 Required Model Adaptations and Parameters

In addition to the query modifications, we also have to slightly adapt our model so that not only queries with explicit temporal and geographic constraints can be processed. In the absence of temporal or geographic constraints, our model assumes that the information need has no temporal or geographic dimension. However, in the GeoTime data set, all queries have at least a latent temporal and geographic aspect (*"where"* and *"when"*). If a query does not contain an explicit temporal constraint and explicit geographic constraint, no s_{temp} and s_{geo} are calculated, respectively. However, we calculate s_{prox} between terms matching the textual query and all temporal and geographic expressions in the documents.

The parameters for s_{text}, i.e., for the BM25 model, are set to standard values ($k_1 = 1.2$ and $b = 0.75$), and the same values are used for s_{temp} and s_{geo}. The α-parameters of Equation 5.21 for weighting the single scores s_{text}, s_{temp}, and s_{geo} are set as follows: if a temporal and a geographic constraint are specified, α_t and α_g are set to 0.2, otherwise, they are set to 0. This is motivated by the intuition that the textual relevance is more important than the temporal and the geographic relevance on its own. If a document satisfies either the temporal or the geographic constraint in addition to q_{text}, it should be considered more relevant than a document not satisfying q_{text} but both the temporal and the geographic constraints. In terms of the GeoTime judgments, the former document would be considered as partially relevant while the latter document would be considered as not relevant (Gey et al., 2010).

In addition, we use different values of β to weight the term proximity feature. This allows to study the influence of this feature, which mainly distinguishes our approach from related approaches. By also using $\beta = 0$, we evaluate our model without the term proximity feature, additionally.

[19] Bounding box information is extracted from GeoNames, `http://www.geonames.org/` [last accessed October 1, 2014].

topic	original query and modified q_{text} query parts for baselines and the full model	q_{geo} and q_{temp}
01	When and where did **Astrid Lindgren die**?	
02	When and where did **Hurricane Katrina** make **landfall** in the United States?	United States
03	When and where did **Paul Nitze die**?	
04	When and where did the **SARS epidemic begin**?	
05	When and where did **Katharine Hepburn die**?	
06	When and where did **anti-government demonstrations** occur in Uzbekistan?	Uzbekistam
07	How old was **Max Schmeling** when he **died**, and where did he die?	
08	When and where did **Chechen rebels** take Russians **hostage** in a **theatre**?	Russia
09	When and where did **Rosa Parks die**?	
10	When was the **decision** made on **siting** the **ITER** and where is it to be **built**?	
11	Describe when and where **train accidents** occurred which had **fatalities** in the period 2002 to 2005.	2002 to 2005
12	When and where did **Yasser Arafat die**?	
13	What **Portuguese colony** was **transferred** to China and when?	China
14	When and where did a **volcano erupt** in Africa during 2002?	Africa; 2002
15	What American **football team won** the **Superbowl** in 2002, and where was the game played?	2002
16	When and where were the last three **Winter Olympics** held?	
17	When and where was a **candidate** for **president** of a democratic South American country **kidnapped** by a **rebel group**?	South America
18	What date was a **country** was **invaded** by the **United States** in 2002?	2002
19	When and where did the **funeral** of **Queen Elizabeth** (the Queen Mother) take place?	
20	What **country** is the most **recent** to **join** the **UN** and when did it join?	
21	When and where were the 2010 **Winter Olympics host** city location **announced**?	2010
22	When and where did a **massive earthquake** occur in December 2003?	2003-12
23	When did the largest **expansion** of the **European Union** take place, and which countries became members?	Europe
24	When and what **country** has **banned cell phones**?	
25	How long after the **Sumatra earthquake** did the **tsunami** hit Sri Lanka?	Sri Lanka

Table 5.2: NTCIR-8 GeoTime topics and queries used for the different models. Underlined parts form q_{text} for BL-text, bold parts form q_{text} for BL-bool and the proximity^2-aware ranking model. For q_{geo}, we show the names of the regions for better readability although we use the respective bounding boxes. For q_{temp}, we use normalized values of the temporal expressions.

5.6.4 Evaluation Results

For our experiments, we use the evaluation metrics precision at k (P@k), average precision at k (AP@k) and normalized discounted cumulative gain at k documents (nDCG@k), which were explained in Section 2.6.2. AP and nDCG have also been used to evaluate the systems of the NTCIR-8 GeoTime participants with maximum k values of 100 (Gey et al., 2010). As mentioned above, some of the documents ranked top-100 by any of our used methods do not have any judgment (neither relevant nor irrelevant) from the GeoTime challenge. In such cases, we set the judgment of those documents to "irrelevant" motivated by the fact that on average, there are almost 700 judgments per topic for all documents, which have been retrieved as relevant by the systems of the GeoTime participants. In addition, our evaluation results are then only determined based on publicly available data and no biased judgments are included.

	precision (P@k)				average precision (AP@k)					nDCG@k				
method	@5	@10	@20	@50	@5	@10	@20	@50	@100	@5	@10	@20	@50	@100
BL-text	44.8	42.0	36.2	29.9	35.6	34.3	27.7	25.6	23.6	45.0	44.8	44.2	46.8	47.2
BL-bool	48.0	44.0	38.6	32.7	40.0	37.6	30.7	29.6	27.7	49.1	47.8	47.2	51.0	52.1
β=0	47.2	42.8	36.6	30.1	39.2	36.8	28.8	26.5	25.6	48.1	46.6	45.1	46.7	48.7
β=0.1	49.6	44.0	39.0	31.2	42.0	38.8	32.2	29.4	28.4	50.5	48.1	47.8	49.3	51.0
β=0.3	**51.2**	45.6	41.4	**33.3**	43.3	39.1	34.3	31.6	**28.9**	52.0	49.5	50.3	52.2	**52.6**
β=0.5	**51.2**	**46.8**	**41.6**	32.9	**44.9**	**40.6**	**34.5**	**31.7**	28.8	**53.2**	**51.1**	**51.2**	**52.9**	**52.6**
β=0.7	49.6	**46.8**	41.4	32.4	43.8	40.1	33.9	30.7	27.8	51.6	50.5	50.5	51.8	51.5
β=0.9	49.6	**46.8**	41.4	32.3	43.6	40.1	34.1	30.7	27.7	51.5	50.5	50.3	51.5	50.9

Table 5.3: Evaluation results on all 25 NTCIR-8 GeoTime topics.

Evaluation Results on the Full Dataset

In Table 5.3, the evaluation results of the two baseline models and of our ranking model with different β-weights for the term proximity feature are presented. Independent of the evaluation metrics and the number of documents (k), the first baseline BL-text is outperformed by the second baseline BL-bool. This directly shows how important it is to consider the semantics of temporal and geographic expressions, i.e., to extract and normalize temporal and geographic expressions and to not consider them as regular terms.

As further important observations, we can see that the proximity^2-aware ranking model outperforms both baseline approaches and that ignoring the term proximity feature, i.e., using $\beta = 0$, decreases the evaluation results independent of the evaluation metrics and the value of k. The best results are achieved with β set to 0.5, i.e., a medium weighting of the term proximity feature. The results demonstrate in particular that the improvements over both baselines are most remarkable when evaluating the top ranked documents ($k = 5$ and $k = 10$). Since the relevance of the top-ranked documents is most crucial for search engines, this shows the importance of taking into account the term proximity between regular terms satisfying q_{text} and expressions satisfying q_{temp} and q_{geo} in addition to considering the semantics of temporal and geographic expressions.

Evaluation Results on Subsets of the NTCIR-8 GeoTime Data

Since the GeoTime topics are very heterogeneous, we further split the results into four groups for a more detailed analysis. In Table 5.4, results for topics with explicit geographic constraints (a), with explicit temporal constraints (b), and with explicit temporal and geographic constraints (c) are presented. In addition, results for topics without explicit constraints are shown in Table 5.4(d).

The differences between the results for topics with geographic constraints and topics with temporal constraints are significant. However, these differences are not due to our model but due to different topic difficulty. There are many more documents judged as relevant for the topics with explicit geographic constraints than for those with explicit temporal constraints (cf. Table 5.1). That is, all topics with explicit temporal constraints were among the most difficult topics in the data set as was also shown in an analysis of the GeoTime organizers (Gey et al., 2010). Despite these differences, on both topic sets, the observations discovered from the whole data set hold for many metrics and k values: (i) BL-bool outperforms BL-text except for small k values on the temporal topics data set, (ii) considering the term proximity feature

(a) Evaluation results on GeoTime topics with explicit geographic constraints.

	precision (P@k)				average precision (AP@k)					nDCG@k				
method	@5	@10	@20	@50	@5	@10	@20	@50	@100	@5	@10	@20	@50	@100
BL-text	65.7	64.3	57.1	46.9	60.6	56.4	42.8	39.2	33.1	68.7	66.9	61.5	64.0	60.5
BL-bool	71.4	67.1	59.3	51.1	63.4	56.5	45.7	44.7	38.8	73.4	69.8	63.9	69.3	66.7
β=0	68.6	64.3	57.9	46.3	58.0	53.2	43.1	41.0	39.0	68.8	66.0	61.4	63.8	64.5
β=0.3	**74.3**	67.1	64.3	**52.6**	63.3	56.1	51.5	**49.9**	**42.7**	73.5	68.9	67.0	**70.9**	**68.4**
β=0.5	**74.3**	**68.6**	63.6	50.9	66.2	**59.0**	50.6	48.2	40.4	**75.0**	**70.7**	67.1	69.6	66.1
β=0.7	**74.3**	**68.6**	63.6	50.0	**66.3**	**59.0**	50.6	47.2	39.3	74.9	70.6	67.0	68.7	64.4
β=0.9	**74.3**	67.1	**65.0**	49.4	65.8	58.0	**53.3**	47.8	39.8	74.7	69.5	**67.9**	68.2	63.5

(b) Evaluation results on GeoTime topics with explicit temporal constraints.

	precision (P@k)				average precision (AP@k)					nDCG@k				
method	@5	@10	@20	@50	@5	@10	@20	@50	@100	@5	@10	@20	@50	@100
BL-text	20.0	16.0	14.0	11.2	6.9	5.7	4.2	4.1	3.8	15.0	13.8	14.2	16.5	16.6
BL-bool	12.0	12.0	**20.0**	16.4	6.0	6.4	11.4	**12.2**	**11.4**	11.5	11.9	18.6	22.9	23.3
β=0	12.0	10.0	12.0	12.0	9.1	7.3	5.8	4.5	4.1	12.8	11.3	12.1	12.0	14.3
β=0.3	20.0	18.0	18.0	16.0	15.1	12.2	10.3	9.3	9.1	21.4	19.5	21.5	23.2	25.2
β=0.5	20.0	22.0	**20.0**	17.2	16.7	14.5	**11.6**	10.7	9.7	22.1	22.9	**23.9**	27.3	26.4
β=0.7	**24.0**	22.0	19.0	17.2	19.9	15.0	11.1	10.8	9.9	24.7	23.1	23.3	27.4	26.6
β=0.9	**24.0**	**26.0**	19.0	**17.6**	**20.7**	**17.1**	9.7	10.4	9.3	**25.0**	**25.8**	23.5	**27.7**	**26.7**

(c) Evaluation results on the GeoTime topic with explicit temporal and geographic constraints.

	precision (P@k)				average precision (AP@k)					nDCG@k				
method	@5	@10	@20	@50	@5	@10	@20	@50	@100	@5	@10	@20	@50	@100
BL-text	20.0	40.0	35.0	18.0	4.0	14.6	10.0	6.0	7.2	13.1	30.6	30.2	27.2	32.3
BL-bool	**100**	**100**	55.0	**36.0**	**100**	**100**	31.5	24.6	30.3	**100**	**100**	68.4	66.2	**77.1**
β=0	**100**	90.0	50.0	22.0	**100**	90.0	27.5	13.9	13.7	**100**	93.6	64.1	50.4	52.0
β=0.3	**100**	**100**	**75.0**	34.0	**100**	**100**	55.3	31.0	30.8	**100**	**100**	82.2	66.4	68.3
β=0.5	**100**	**100**	**75.0**	**36.0**	**100**	**100**	**55.7**	**33.2**	**31.7**	**100**	**100**	**82.3**	**68.6**	68.6
β=0.7	**100**	**100**	**75.0**	**36.0**	**100**	**100**	54.8	32.5	31.1	**100**	**100**	**82.3**	68.5	68.5
β=0.9	**100**	**100**	**75.0**	**36.0**	**100**	**100**	54.5	32.2	30.7	**100**	**100**	82.2	68.5	68.5

(d) Evaluation results on the GeoTime topics without explicit temporal or geographic constraints.

	precision (P@k)				average precision (AP@k)					nDCG@k				
method	@5	@10	@20	@50	@5	@10	@20	@50	@100	@5	@10	@20	@50	@100
BL-text	45.0	**40.0**	33.3	28.8	35.6	35.0	30.1	28.3	27.7	46.4	46.1	47.8	50.9	53.5
BL-bool	45.0	39.2	32.9	28.5	35.6	34.5	30.0	28.4	27.9	46.4	45.6	47.5	50.7	53.4
β=0	45.0	**40.0**	33.3	28.8	35.6	35.0	30.1	28.3	27.7	46.4	46.1	47.8	50.9	53.5
β=0.3	**46.7**	**40.0**	**35.0**	**29.2**	38.6	35.3	32.5	30.2	29.0	48.2	46.5	49.9	52.2	53.4
β=0.5	**46.7**	**40.0**	**35.0**	28.7	**39.6**	**35.8**	**33.0**	**30.7**	**29.8**	**49.4**	**47.3**	**50.8**	**52.6**	**54.2**
β=0.7	41.7	**40.0**	**35.0**	28.2	35.9	34.6	31.9	29.3	28.4	45.1	46.1	49.5	50.8	52.9
β=0.9	41.7	39.2	34.2	28.2	35.6	34.3	31.3	29.1	28.0	45.0	45.5	48.5	50.2	52.1

Table 5.4: Evaluation results on subsets of the GeoTime topics.

(i.e., using $\beta > 0$) improves evaluation results, and (iii) the proximity^2-aware ranking model with a medium β-weight outperforms both baselines in particular for the top-ranked documents ($k = 5$, $k = 10$). The improvements for the top ranked documents are particularly remarkable for the topics with explicit temporal constraints.

In Table 5.4(c), the results for topic GeoTime-14 are presented – the only topic with explicit temporal and explicit geographic constraints. The huge differences between the two baselines show how important it is to include the semantics of temporal and geographic expressions into the ranking model when dealing with explicit temporal and geographic constraints. While the baseline with boolean constraints already achieves very good results for top ranked documents, the results for more documents ($k \geq 20$) can further be improved by integrating proximity information into the model. The best evaluation results are again achieved with a medium β-weighting of 0.5.

After having demonstrated the value of our proximity^2-aware ranking model for explicit temporal and/or geographic expressions, we finally analyze if proximity information also helps to improve the retrieval quality when being faced with implicit temporal and geographic constraints only (*"when"* and *"where"*). As Table 5.3(d) shows, our ranking model with $\beta = 0.5$ achieves the best results on this subset and outperforms the baselines. Note that the proximity^2-aware ranking model with $\beta = 0$ is identical to the BL-text baseline since only a text score is calculated.

Summary

The evaluation results mostly confirm the three hypotheses formulated at the beginning of Section 5.6 so that we can summarize the evaluation results with the following three main observations:

- Considering the semantics of temporal and geographic expressions helps to improve satisfying information needs with explicit and/or implicit temporal and geographic constraints.
- Taking into account term proximity information between regular terms satisfying a text query and expressions satisfying temporal and geographic queries helps to improve ranking results. In particular, the top-ranked documents benefit from term proximity information.
- Combining textual, temporal, and geographic relevance scores outperforms a standard textual ranking with boolean temporal and geographic filtering – however, only if proximity information is also considered. Relying solely on the combination of textual, temporal, and geographic relevance scores results in partially better and partially worse evaluation numbers than BL-bool. Note, however, that many more documents retrieved with the proximity^2-aware ranking model than with the baseline models had no judgment. Thus, documents without judgments should be checked to finally answer the second hypothesis. However, the most important hypothesis, i.e., the value of considering term proximity information, could be demonstrated without collecting further relevance judgments.

When analyzing the search results after the evaluation, we made the following observation: Since all documents are from the New York Times corpus and thus news documents, the document creation time (DCT) plays an important role throughout the whole text of the documents (cf. Section 3.3). Thus, if the DCT satisfies q_{temp}, the temporal constraint could be softened for the term proximity calculation since the DCT is latently available in the whole news article. However, although we evaluated our newly

developed model based on a dataset with news articles only, we did not develop the proximity^2-aware ranking model specifically for news documents but aim at applying the model to narrative-style document collections, e.g., to develop a spatio-temporal search engine for Wikipedia. Thus, we did not include such domain-specific adaptations.

5.7 Summary of the Chapter

Standard approaches in information retrieval do neither treat temporal and geographic expressions in documents in a special way nor do they allow for formulating meaningful temporal and geographic constraints. However, temporal and geographic information needs are quite frequent and there is a need to better serve them. In this chapter, we addressed the topic of spatio-temporal information retrieval. After a survey of related work in the areas of temporal information retrieval, geographic information retrieval, and spatio-temporal information retrieval, we first developed a multidimensional query model, which allows to combine a textual query with well defined temporal and geographic constraints.

As further major contribution of this chapter, we developed an approach to spatio-temporal information retrieval, namely the proximity^2-aware ranking model. In contrast to previous works addressing temporal information needs, geographic information needs, or both, our model does not consider the single parts of a multidimensional query as being independent of each other. To eliminate the independence assumption, our model does not only combine textual, temporal, and geographic scores for the final ranking, but also considers a term proximity score. This score is computed based on the textual distance of terms satisfying all parts of the query. A second main feature of our model is that temporal and geographic proximities between temporal and geographic expressions in the query and expressions in the documents are considered to allow a meaningful ranking of documents not fully satisfying all query parts.

In addition to describing efficient indexing and query processing strategies, we then performed an evaluation that demonstrated in particular the following two aspects: (i) It is important that temporal and geographic expressions are not treated as regular terms. In contrast, their normalized semantics should be considered to decide if a document satisfies temporal and geographic information needs. (ii) Taking into account term proximity information between terms satisfying all three query dimensions significantly improves the retrieval performance.

While we performed our evaluation on a publicly available data set with a news corpus as underlying document collection, for future work, an evaluation on narrative-style documents such as Wikipedia would be desirable. In addition, the queries should be natively formulated as multidimensional queries containing a textual part and explicit temporal and geographic constraints. However, setting up such an evaluation scenario is probably only feasible in the context of a cooperative project, e.g., by organizing an evaluation campaign such as the GeoTime competitions.

6 Event-centric Search and Exploration

While we addressed spatio-temporal information retrieval in the previous chapter, we now exploit explicitly combined geographic and temporal information, i.e., spatio-temporal events, and introduce several event-centric search and exploration approaches. After we further motivate these tasks by describing the objectives and outline of this chapter, we briefly survey in Section 6.2 some related approaches on exploiting automatically extracted combinations of geographic and temporal information or event concepts, which are similar to our spatio-temporal events introduced in Chapter 4. Starting with Section 6.3, we present our approaches to event-centric search and exploration.

6.1 Motivation and Objectives

Our claim is that for many categories of documents, events are essential to describe a topic or theme. This holds, among others, for documents about history or biographies, an aspect already introduced in Chapter 4 and which we will elaborate on later in this chapter.

In the previous chapter, we developed a spatio-temporal search and retrieval approach allowing for querying document collections with respect to textual, temporal, and geographic constraints. However, when being interested in an event-centric exploration of document collections, it is more intuitive to directly search for events rather than for geographic and temporal information separately. In Section 6.3, we will thus describe approaches to event-centric information retrieval – with either returning as search results a ranked list of documents or an (ordered) list of events occurring in the documents of the queried document collection. Furthermore, we describe how search results of event-centric information retrieval can be presented and explored in map-based scenarios in Section 6.4.

Another important task in the context of exploring document collections is to identify similar documents, and to allow a user to jump from an initial document to the most similar ones. Obviously, determining similarity is a subjective matter and documents can be similar with respect to multiple aspects, such as words, length, or more complex semantic concepts. In Section 6.5, we develop a similarity model for identifying event-centric document similarity solely relying on extracted spatio-temporal events. This model will be term- and language-independent, because the similarity between documents is calculated on normalized event information. Being able to identify event-centric similarity between documents allows for an efficient way of event-centric exploration of document collections, and due to the language independence of the model, document similarity can even be detected across documents of different languages.

In Section 6.6, we briefly outline how the developed event-centric document similarity model can be extended to determine event-centric similarity between persons, again solely relying on extracted information from large text collections. Furthermore, we present an adaptation of our document similarity model to another text genre so that not spatio-temporal events but biomedical events are used for determining document similarity. Finally, we will summarize the chapter in Section 6.7.

6.2 Related Work on Event-centric Information Retrieval & Exploration

An overview of related work on temporal, geographic, and spatio-temporal information retrieval was already provided in Chapter 5. In that overview, the focus lied on approaches mainly addressing the querying and document retrieval process for queries with temporal constraints, geographic constraints, or both. In contrast, in this section, we present some related approaches that focus on the exploration of combined spatio-temporal information. However, since temporal and geographic information is also used in combination in the research area of topic detection and tracking (TDT), and since it also deals with events, we first distinguish our goals from those of TDT. Finally, we briefly explain the task of entity-oriented search since it is quite similar to our event-centric search which results in retrieving events instead of documents as search results.

Topic Detection and Tracking

An area related to our work is topic clustering and in particular topic detection and tracking (TDT) where items of a document stream (e.g., a news stream) are analyzed. The goals of these approaches are to detect new unreported news events, and to track topics by assigning documents to already detected events (Allan, 2002a). There is a lot of research dealing with TDT for which the identification of events is necessary. Often, the similarity measures use information of named entity recognition, e.g., locations, temporal expressions, or person and organization names mentioned in the documents (Makkonen et al., 2003; Zhang et al., 2007). However, in contrast to our work, TDT systems try to identify a main event that can be associated with documents. Our goal, on the other hand, is to identify as many events in documents as possible, and to use the identified events for event-centric search and exploration tasks. Thus, the event concept in TDT differs from our spatio-temporal events (cf. Section 4.2).

Approaches to Combined Spatial and Temporal Information Exploration

While some of the works surveyed in the previous chapter also discussed map-based exploration of search results and are thus also similar to some aspects we will present in this chapter, Mata and Claramunt (2011)'s approach to spatio-temporal information retrieval could have also been described there. However, it explicitly combines geographic and temporal information for exploration purposes.

In their approach that "is based on a [semi-automatically built] domain ontology that integrates the geographic, temporal and thematic dimensions related to some [...] objects of interest" (Mata and Claramunt, 2011), the objects of interest are either geographic entities or events. Thus, the temporal and the geographic dimensions are often coupled as attributes to events, which makes their approach quite similar to our event-centric search and exploration scenarios. Furthermore, for the exploration of search results, they also describe the idea to display a temporally ordered sequence of events on a map "depicted by an oriented line that connects the locations" (Mata and Claramunt, 2011). Although this exploration functionality is similar to our concepts of *document trajectories* and *event sequences* that we will present in Section 6.4 and which we initially introduced in (Strötgen et al., 2010; Strötgen and Gertz, 2010b), their events have to exist in the pre-defined ontology, which is a major limitation.

Wang et al. present an approach to "spatio-temporal knowledge harvesting" (Wang et al., 2011) with the goal of constructing trajectories of individuals, however, without rich search and exploration support. From a news corpus, person, location and time entities are extracted. While the disambiguation of person names and locations is briefly explained, no information about normalizing temporal expressions is

provided. Spatio-temporal facts (person, location, time) are generated if the three expressions occur within a sentence and if they hold against the pruning rules (preposition required if location before person; verb required between person and location if person before location). To build each person's trajectory, "the facts are grouped by person first, and sorted according to the time" (Wang et al., 2011).

While the idea is again quite similar to our concepts of *document trajectories* and *event sequences*, no information is provided how the trajectories are generated if different granularities of temporal expressions and geographic expressions occur – an issue that we will address in Section 6.4. Furthermore, on a corpus of 1.8 million news documents, less than 80,000 facts are extracted, "which indicates that the overall recall is not high" (Wang et al., 2011).

Similarly, the idea to express biographies as sequences of events in space and time and to display them on a map was also presented by Gey et al. (2008). Focusing on historical biographies and defining "a biography as ordered sequence of events [with] an event [being] [...] a 4-tuple (Action, Date-range, Place, Other people)" (Gey et al., 2008), the authors point out that "unanticipated and seemingly unrelated connections" (Gey et al., 2008) can be discovered between different persons. Although it is stated that named entity recognition is applied, the system description is unfortunately rather short and no further details are provided how the biographies are extracted or finally created.

An approach to the extraction and geographical navigation of important historical events extracted from Japanese text documents is described by Yamamoto et al. (2011). While the event extraction process starts with collecting Web pages relevant to a specific person or happening, the detected events are then assigned extracted temporal and geographic information and form the basis of a so-called "'Virtual History Tour', which achieves the navigation of the extracted events on a map interface with automatic movements in chronological order" (Yamamoto et al., 2011).

In all the described works, combined geographic and temporal information extracted from multiple documents can be explored together on a single map. Thus, the principle is to move from documents to events as the search result to a query – an idea that we will also realize when presenting our approaches to event-centric search and exploration. In general, this idea to retrieve entities instead of documents when querying a document collection for entities is addressed in entity-oriented search.

Entity-oriented Search

The main idea of research on entity-oriented search is to put the entities into the center of interest and provide entities as search results instead of documents (Balog et al., 2012). The motivation for entity-oriented search is that "[m]any user information needs concern entities, [e.g.,] people, organizations, locations, products" (Balog et al., 2012). In such cases, the information need is best addressed if search results are the specific entities (with further, linked information) and not documents in which the entities occur. In this chapter, the idea of returning events and functionality to explore them play a central role.

6.3 Event-centric Search

The idea behind performing event-centric search instead of spatio-temporal search is that given a user information need consisting of topical information represented as q_{text} and temporal and geographic constraints for the events represented by temporal and geographic query parts q_{temp} and q_{geo}, the

document collection is searched for events satisfying the temporal and the geographic constraints. Thus, while the querying procedure is quite similar to the one presented in Chapter 5, the retrieval task is different. Not the temporal and geographic document profiles, but the event document profiles of a document collection have to be analyzed to detect relevant events and documents.

A second difference to spatio-temporal information retrieval is that other types of search results are also suitable instead of returning a ranked list of documents. While documents can also be returned in a meaningful way – e.g., ranked by topical relevance, the number of relevant events, or a combination of it – a list of events instead of documents can be returned similarly as in entity-oriented search. In Section 6.3.3, we present a ranking strategy for returning a list of documents containing relevant spatio-temporal events, and in Section 6.3.4, we describe a procedure to return lists of events instead of documents. Before that, however, we introduce a concept to concisely organize events extracted from multiple documents of a document collection (Section 6.3.1) and the concept of so-called *event snippets*, which can be used to demonstrate the relevance of single search results to the user (Section 6.3.2).

6.3.1 Cross-document Event Sets

Putting events into the center for the task of document collection exploration, we require a concept to deal with events extracted from different documents. Thus, we introduce so-called *cross-document event sets*:

Definition 6.1. *(Cross-document Event Set)*
Given a document collection D, all events extracted form documents $d_i \in D$ are put together in a *cross-document event set,* $ces(D)$.

Since each spatio-temporal event in $ces(D)$ is a tuple of the form $\langle te_i, ge_j \rangle$, and since each extracted temporal expression te_i and each extracted geographic expression ge_j contain document and offset information in addition to normalization information (cf. Definition 4.2 and Definition 4.3, page 136), each event extracted from any $d_i \in D$ is unique. Note, however, that also each event in any $ces(D)$ is unique, we consider events with identical normalized temporal and geographic information as *same events* in the following, i.e., as identical events occurring in different documents or at different positions in a document.

To build $ces(D)$, the events of the event document profiles of all $d_i \in D$ can directly be accessed. In addition, temporal and geographic constraints can be defined to include only a subset of the events occurring in the documents of D or the set of considered documents can be limited. Cross-document event sets become important when returning events as search results.

6.3.2 Event Snippets

Similar to standard Web snippets as provided by all major search engines, we also use the concept of snippets to directly provide information why a document or an event is determined as relevant for an event-centric query. Instead of highlighting standard query terms, the temporal and geographic expressions forming the relevant event are highlighted in their textual context, e.g., by showing the sentence from which the event is extracted.

Furthermore, the snippets also should contain normalized temporal and geographic information. This is particularly valuable if underspecified or relative temporal expressions or ambiguous geographic expressions form the event since only limited textual context can be shown in a snippet and such expressions are

often difficult to interpret without sufficient context information. For instance, if the expressions "March 11" and "Heidelberg" form an event and are highlighted in an event snippet, normalized information such as "March 11, 2014" and "Heidelberg (South Africa)" is very valuable for the user. Note that instead of using standard normalization values, fully specified temporal and geographic information can be shown for better readability, e.g., "March 11, 2014" instead of "2014-03-11". Examples of event snippets will be presented below.

If documents are the main search result and if the same event is extracted more than once from a document, simple heuristics about the event information can be used to rank the sentences containing the events and only the top ranked sentence is shown by default. For instance, the importance of temporal expressions in a document with respect to a query can be determined to rank the sentences.[1] Of course, the user can also decide to visualize and explore the events extracted from the other sentences as well.

Similarly, in the case of events being the main search result and if an event was detected in multiple documents, an event snippet containing the event references to the most important documents can be created. How to determine the most important documents is explained in the next section. Note that also if multiple event references are presented, the normalized event information has to be shown only once since it is obviously identical for all event references.

6.3.3 Retrieving Relevant Documents

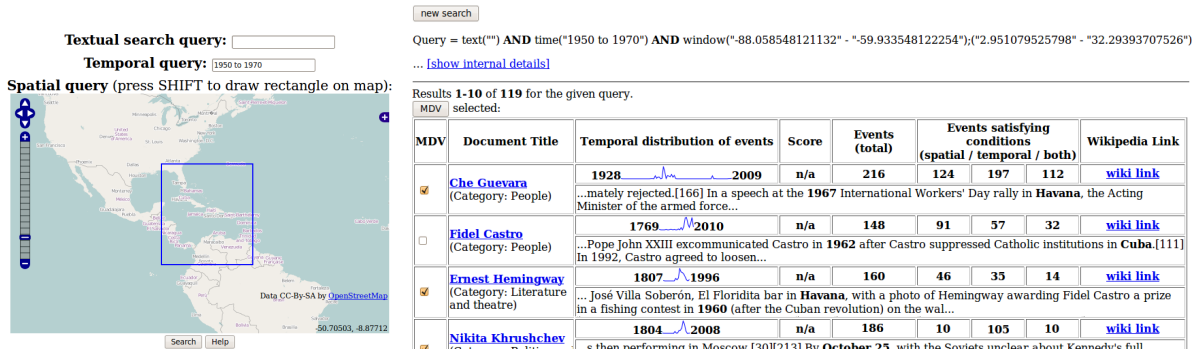
In our first prototype (TimeTrails) to explore spatio-temporal information extracted from a document collection D in an event-centric way (Strötgen and Gertz, 2010b), we provide four ranking possibilities for the hit list of documents returned for a query $q = \{q_{text}, q_{temp}, q_{geo}\}$. The first ranking is based on the relevance scores retrieved for the textual part of a query q_{text}. In the implementation, we used Lucene's standard relevance score.[2]

As second and third ranking methods, we determined for each document $d_i \in D$, the number of events satisfying the temporal constraint or the geographic constraint, respectively. Thus, the documents in the result list can be ordered by the counts of events satisfying either q_{temp} or q_{geo}. Finally, as fourth ranking method, we combine the temporal and the geographic constraints and order the documents based on the number of events satisfying both q_{temp} and q_{geo}. Obviously, if only one or two query parts are specified, not all orderings are possible.

In Figure 6.1, we present an example search scenario using the TimeTrails system. For the query presented in Figure 6.1(a), a screenshot of TimeTrails' search result user interface is depicted in Figure 6.1(b). Since the underlying corpus consists of a subset of Wikipedia with a special focus on biographies and documents about history, the results for the search with the temporal constraint "1950 to 1970" and the geographic constraint "Caribbean region / Central America" (specified as a rectangle on the map) contains Wikipedia articles about "Che Guevara", "Fidel Castro", "Ernest Hemingway", and "Nikita Khrushchev" as top ranked documents.

[1] For instance, we developed in the context of a research cooperation a model to determine which temporal expressions in documents are the most important ones, either in general or with respect to a query (Strötgen et al., 2012b). This model, which relies on expression, document, corpus, and query features, can be used to determine which reference to an event should be selected to be displayed as event snippet.

[2] Lucene, `http://lucene.apache.org/` [last accessed Juli 28, 2014].



(a) Query interface. (b) Search result user interface.

Figure 6.1: Timetrails' query interface (a) and its search result user interface (b).

Scenarios regarding the exploration of the search results will be detailed in Section 6.4 when presenting several map-based exploration approaches. Thus, some of the features shown in Figure 6.1(b) will be explained there. In addition to such features, with each document in the hit list, the document's title, its category, and an event snippet is provided. Furthermore, the total number of events extracted from the document is shown as well as the counts of events satisfying the geographic, the temporal, and both constraints. Finally, the range and number of temporal expressions occurring as part of events in the document are also visualized in the form of a sparkline, i.e., a simple, word-sized graphic with high data density (Tufte, 2006). This information already gives the user a good idea of the amount of events identified for each document, as not for all documents a spatio-temporal exploration is equally meaningful.

6.3.4 Retrieving Relevant Events

Retrieving relevant events instead of relevant documents as search results is a powerful way to allow for an event-centric exploration of document collections. As concept for organizing events extracted from different documents, we rely on cross-document event sets introduced in Definition 6.1.

For our prototypical event search engine (Strötgen and Gertz, 2012a), we select meaningful events that are to be shown as results given a query $q = \{q_{text}, q_{temp}, q_{geo}\}$ and a document collection D. Since not all query parts have to be specified, different strategies have to be applied depending on which parts of a query are provided. In the first case, we assume that the textual part of a query is specified, i.e., q_{text} is not empty, and thus run the following strategy:

1. Rank documents of D according to q_{text} using a standard textual relevance measure, e.g., BM-25 that was described in Section 5.4.2.

2. Select top-k documents that have at least one event satisfying q_{temp} and q_{geo} – if q_{temp} and q_{geo} are specified. Whether documents contain events matching the query parts can efficiently be determined using the event document profiles as well as the temporal and geographic indexes explained in detail in Chapter 4 and Chapter 5.

3. Combine all events from the top-k documents' event document profiles that satisfy q_{temp} and q_{geo} and store the events in a cross-document event set.

Thus, in this case, the assumption is that the most relevant documents with respect to q_{text} also contain relevant events. In the presence of q_{temp} and q_{geo}, the events and the corresponding documents are further constrained.

In the second case, we assume that the textual part of a query is not specified, i.e., either q_{temp}, q_{geo}, or both are provided but q_{text} is empty. Then, relevant events are solely retrieved based on the conditions q_{temp} and q_{geo}.

1. Rank documents of D according to the number of events satisfying q_{temp} and q_{geo}. This again can be done efficiently based on the index structures for the event components.
2. Select top-k documents from the ranked list.
3. Combine the event document profiles of these k documents into a cross-document event set, considering only events that satisfy q_{temp} and q_{geo}.

By using only the top-k relevant documents for creating the event sets, only the events of those documents are taken into account that are most relevant to a specific subject defined by q_{text} or that are most relevant for the specified temporal and geographic constraints, respectively. Thus, we avoid events being part of a cross-document event set that only occur in irrelevant documents.

Returning Events

The next question arising is how to return the set of events. For this, there are multiple options: (i) events can be returned as a list ranked by the number of how often they occur in the cross-document event set; (ii) events can be chronologically ordered based on their normalized temporal information; (iii) events can be clustered based on their granularities, e.g., all events with a rather fine temporal granularity can be clustered in year buckets; (iv) events can be clustered based on both their temporal and their geographic granularities; and (v) events can be visualized on a map based on their normalized geographic information. Furthermore, a map-based view can be combined with any of the temporally ordered lists.

In our prototype, we use such a combination of a map with a chronologically ordered and granularity-clustered list and describe the details in the next section where we present several map-based exploration scenarios.

6.4 Map-based Exploration

In the TimeTrails system, several map-based exploration scenarios are realized and they can be accessed from the search result user interface (cf. Figure 6.1). These are presented in the following. Additionally, some further exploration scenarios are described that are particularly useful if event sets are directly returned as search results instead of documents.

6.4.1 Document Trajectories and Event Sequences

If documents are returned as search results for an event-centric query, the events of single documents are to be visualized, i.e., events of interest are organized in an event document profile. In contrast, if events are directly returned as search results, the events of multiple documents and thus cross-document event sets are to be visualized. Both, an event document profile and a cross-document event set are just

sets of events. Based on their geographic component, in general, events can easily be visualized on a map. However, it might be hard to see relationships among events on a map. Thus, given the temporal component of an event, it seems natural to organize events chronologically and to choose another form of representation for a set of events. For this, in our approach we use the concepts of *document trajectories* and *event sequences* for event document profiles and cross-document event sets, respectively.

Conceptually, a document trajectory corresponding to an event document profile and an event sequence corresponding to an event set are chronologically sorted sequences of events. Analogous to trajectories of moving objects extensively studied in the area of moving object databases, such a document trajectory or sequence can be considered a path a subject is taking over time.

In the case of a document trajectory or event sequence, such a path is based on the locations corresponding to where the events extracted from an event set "happen" and is sorted chronologically. This is an intuitive approach even for several types of documents such as biographies or historical documents where in a document events are described, but not necessarily in a chronological order. Given an event set (from a single document or collection of documents), it is not trivial to construct a document trajectory or an event sequence. For this, we have to distinguish different cases that depend on the granularity of the normalized values of the temporal component of the events.

The most simple case is when all normalized values have the same granularity. Then, based on these values, a complete order can be determined among the events. What still can happen is that for two temporal expressions te_i and te_j of two events e_i and e_j, the normalized values $v(t)_i$ and $v(t)_j$ are identical. In the case where the two events have been extracted from the same document, we assume that the event with the smaller offset $p(t)$ precedes the other event, i.e., that an earlier mentioned event precedes a later mentioned event. For example, for the two events ⟨2011-01-15, Berlin⟩ and ⟨2011-01-15, Hamburg⟩, if the expression corresponding to the first event occurs after the expression for the second event in the document, then the event with location "Hamburg" would precede the event with location "Berlin" in the event sequence. If the two events have been extracted from different documents, using the offsets of the events is not meaningful. In such a case, we assume that there is an order among the documents – namely a relevance ranking – and the order among the events then follows that document order.

The more difficult case is when the normalized temporal values of the events are of different granularity. Assume, for example, the four events $e_1 = \langle$2011-02-10, Hamburg$\rangle$, $e_2 = \langle$2010-12-12, Vienna$\rangle$, $e_3 = \langle$2011-02, Munich$\rangle$, and $e_4 = \langle$2011-02, Heidelberg$\rangle$. Clearly, one cannot immediately establish a total order among these events. Our approach to handle such cases is as follows. First, we build groups of temporally ambiguous events. In the above example, such a group comprises e_1, e_3, and e_4. For each group, we map the normalized values to the coarsest temporal granularity in that group (here the type month) and then establish an order among these events based on the document order as described above. For example, assume a list of documents d_1, d_2, and d_3 with event sets $\{e_4\}, \{e_1, e_3\}$, and $\{e_2\}$, respectively, where in d_2, e_1 occurs before e_3. We then obtain the following order among the events: e_2, e_4, e_1, e_3.

An event set and its corresponding document trajectory or event sequence can thus easily be determined for a document or set of documents, respectively, and thus is accessible to event-centric exploration tasks as detailed in the following.

6.4.2 Single Document Visualization

Using the TimeTrails system, the search results for an event-centric query are first returned as a ranked list of documents with a range of further information as explained above and depicted in Figure 6.1(b). In the first exploration scenario, the user selects a document from the hit list for a map-based visualization. This single document visualization (SDV) can be accessed from the search result user interface through the document titles in the hit list. TimeTrails then visualizes the trajectory of this document on the map, i.e., all geographic locations that occur in the event document profile of the selected document are shown on the map, connected by directed lines, representing the document trajectory, i.e., the temporal relations between the individual locations.

Alternatively, the visualization can also be restricted to those events of the event document profile that satisfy the temporal and geographic constraints of the user query. In particular if a document contains many events, this helps the user to focus on those events mentioned in the document that satisfy the user's initial information need.

6.4.3 Multiple Document Visualization

In TimeTrails' second exploration scenario, the user can select multiple documents from the hit list to explore them simultaneously in the multiple document visualization (MDV) view. Note that the MDV-view displays the document trajectories of the selected documents at once, i.e., multiple trajectories are visualized in the same way as in the SDV-view. Since TimeTrails' search results are document-based, the events of the respective documents are not combined into a cross-document event set and thus not visualized as a single event sequence.

An example of the MDV-view is depicted in Figure 6.2. Here, the document trajectories of three documents of the hit list shown in Figure 6.1(b) are displayed. On the right side of the figure, each document trajectory is listed and information about the events of each document can be explored. In addition, the *intersections* of the document trajectories are listed, which will be further detailed below. Furthermore, the spatio-temporal event "1962-10, Cuba" has been selected so that the respective event snippet is also visualized with two references to the event – one occurring in the document "Che Guevara" and one in the document "Nikita Khrushchev".

Similar to the multiple document visualization of the TimeTrails system, event sequences can be visualized with our prototypical event search engine (Strötgen and Gertz, 2012a). In Figure 6.3, a screenshot of the search result for a query similar to the one used for the TimeTrails system is shown in the form of an event sequence. As explained in Section 6.3.4, the cross-document event set contains events of the top-k documents, and in the example, k is set to 5. In addition, on the right side next to the map, further information about the documents, the events, and the event sequence are listed. More specifically, at the top of the list, the documents contributing to the event sequence are presented, followed by an overview of so-called multi-document events. These correspond to intersections of document trajectories and are explained below.

Next, four types of event sequences are listed and the user can select the one that shall be displayed on the map. The "multi-document event sequence" contains only events that are extracted from more than one document. This results in a clearer representation of the most important events. Alternatively, event sequences containing events of different temporal granularities ($v(t) \leq$ year, $v(t) \leq$ month,

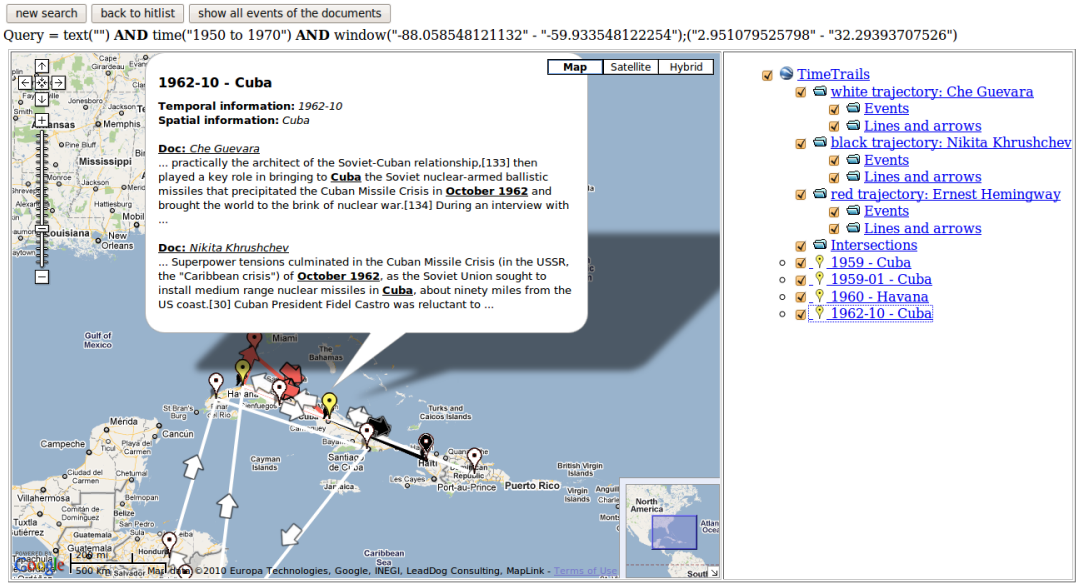


Figure 6.2: Map-based visualization of TimeTrails' search results with three document trajectories.

or $v(t) \leq$ day) can be selected. In the figure, the event sequence with all events for which $v(t) \leq$ day holds true is shown.

6.4.4 Event Snippets on a Map

While *event snippets* have already been introduced in Section 6.3.2, they first were used in the document hit list of the TimeTrails system (cf. Figure 6.1(b)). However, they can also be valuable in the map-based exploration scenarios. To directly get more detailed information about events shown on a map they are visualized if a user selects a specific location or an event of the event listed next to the map. As shown in Figure 6.2 and Figure 6.3, they contain the geographic and the temporal information of the event, the document's title and the sentences containing the event description.

6.4.5 Intersecting Document Trajectories and Multi-document Events

Finally, a last feature for an event-centric, map-based exploration of search results are intersecting document trajectories and multi-document events. In the case of events being the main search result, trajectories of two or more documents can intersect at a location due to two reasons. First, events with identical location information occur in the documents but have different temporal information. Second, the events at that location can happen at the same time, i.e., the same event is mentioned in the documents. Such same events are called multi-document events in the context of event sequences visualizing events of cross-document event sets.

Whenever intersecting document trajectories or multi-document events occur, they are hints that two or more documents are similar with respect to the events they are containing. Thus, a further event-centric exploration scenario for (large) document collections is to determine event-centric similarity between all documents. This idea is addressed in the following section as final major contribution of this thesis.

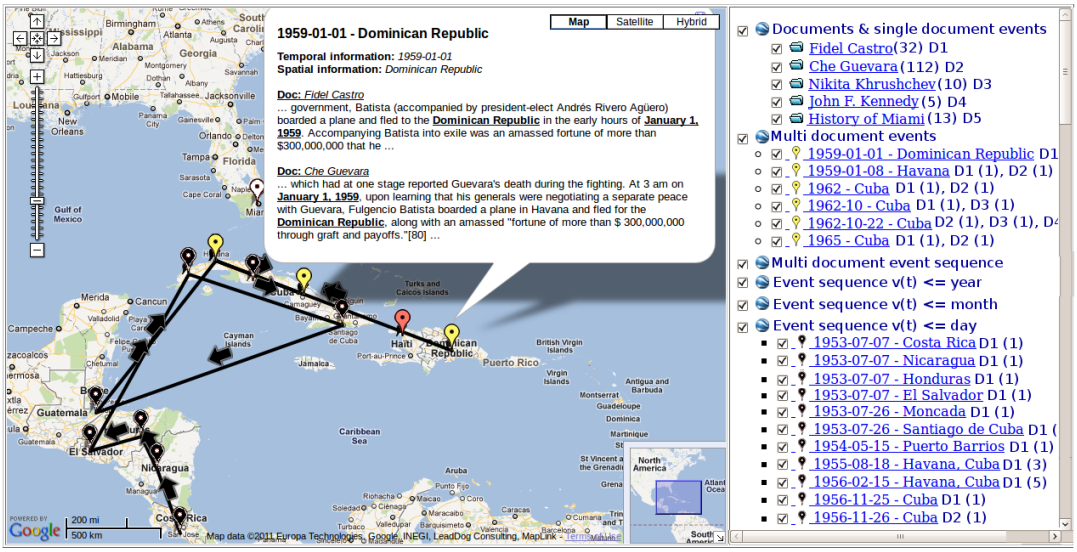


Figure 6.3: Visualization of an event sequence for the top-5 documents of an event-centric search.

6.5 Event-centric Document Similarity

Document similarity measures play an important role in many document retrieval and exploration tasks. However, when are two documents similar? In general, this is a quite subjective and application-specific question, and documents can be similar, amongst others, according to the words they contain, their language, their structure, the topic they address, or the semantic concepts that have been associated with the documents. This plethora of different similarity aspects clearly does not lead to a single, universally applicable document similarity measure. Instead, different measures may lead to new insights into document similarity that cannot be captured by just one single approach. As will be discussed in Section 6.5.1, over the past decades, several models and techniques have been developed to determine a ranked list of documents similar to a given query document. Interestingly, many existing approaches rely on extensions to the vector space model and are rarely suited for multilingual corpora.

In this section, we present a document similarity measure that is based on spatio-temporal events extracted from documents. Using the event-related concepts introduced in Chapter 4, namely event document profiles and the algorithms to compare the temporal and geographic components of two events with each other, documents will be compared and ranked solely relying on event information. Thus, one key feature of our event-centric document similarity model is that two documents can still be determined as similar even though the documents do not describe exactly same events. Another key feature of the model is that it is term- and language-independent. Therefore, documents written in different languages can be compared so that similar documents can be determined across languages, an important feature in the context of document exploration. Finally, exploring event-based similarity among documents may also lead to new information that was not explicit before. For example, even though two documents talk about completely different topics, they both may mention same events. This new information then can be used to investigate and establish new cross-document references.

After briefly presenting relevant related work on document similarity research in the next section, we define the problem statement for the task of calculating event-centric document similarity in Section 6.5.2.

Then, we introduce our measure for calculating the similarity between two single events and develop the event-centric document similarity model itself in Section 6.5.3 and Section 6.5.4, respectively. In Section 6.5.5, we explain how event-centric document similarity can be computed efficiently. Finally, we describe the evaluation settings (Section 6.5.6) and corpora (Section 6.5.7), and demonstrate the effectiveness of the model in Section 6.5.8 by performing experiments on different (multilingual) corpora.

6.5.1 Related Document Similarity Measures

There are many approaches to the computation of document similarity in different IR related tasks such as document classification and clustering. As already mentioned in Chapter 5, in standard information retrieval, documents are typically represented as vectors, as are the queries (Manning et al., 2008: p.113; Baeza-Yates and Ribeiro-Neto, 1999: p.27). These vectors consisting of weights, such as tf-idf, for all terms in the documents are then used to calculate the similarity between a query and a document or between two documents. An example method to determine the similarity between two document vectors is to use the cosine similarity (see, e.g., Manning et al., 2008: p.111).

Obviously, there are numerous approaches to improve such standard similarity measures. For instance, Chim and Deng (2008) use phrase-based document similarity for clustering and show that feature vectors of phrase terms can be seen as expanded feature vectors for single-word terms. Furthermore, latent semantic analysis (Deerwester et al., 1990) was extensively used to improve determining document similarity (see, e.g., Brants and Stolle, 2002), while other techniques employ explicit knowledge representation schemes such as ontologies to estimate semantic relatedness between documents (see, e.g., Gabrilovich and Markovitch, 2007).

Somewhat similar to our approach of applying concept hierarchies to temporal and geographic information, Lakkaraju et al. (2008) use general concept trees to classify documents according to a taxonomy. Our concept hierarchies, however, are very small and specific to temporal and geographic aspects. There is also related research on similarity for event identification in social media (Becker et al., 2010), however, in this study general types of events are considered so that the work relies on another definition of the event concept.

Finally, in the area of cross-language information retrieval, similarity calculations are performed on multilingual corpora. These calculations are usually based on translations while our approach uses normalized, language-independent information. There is only very few work on multilingual, translation-independent document similarity. One approach that makes use of a multilingual thesaurus for computing similarity has been proposed by Steinberger et al. (2002). By relying on thesaurus descriptor terms automatically assigned to documents, their document similarity model does not rely on terms occurring in documents or on their translations but solely on normalized information, i.e., the similarity between documents is independent of the documents' languages. While Steinberger et al. assign all kinds of normalized thesaurus information, we focus on determining event-centric similarity and use solely normalized temporal and geographic information.

An interesting empirical evaluation of models for text document similarity was conducted by Lee et al. (2005). They conclude that many automatic models have very good precision but poor recall, and that "the best performed models [...] are able to detect only a subset of the highly semantically similar document pairs" (Lee et al., 2005). This observation is a motivation for our approach, because we do not want to replace existing similarity measures, but we do want to provide a measure for non-standard document similarity to identify new information, that is, event-based similarity relationships between documents.

6.5.2 Problem Statement

For our event-centric document similarity model, we assume that all documents of a document collection are represented through their event document profiles as they were formally defined in Section 4.4.4 (Definition 4.12, page 146). Thus, the problem statement for calculating event-centric document similarity can be formulated as follows:

> Given two documents d_1 and d_2 and their event document profiles $edp(d_1)$ and $edp(d_2)$, the event-centric document similarity $d\text{-}sim_e(d_1, d_2)$ should be determined in a concise and meaningful way based on the similarity between all events in $edp(d_1)$ and those in $edp(d_2)$.

Obviously, a fundamental step for calculating event-centric document similarity is to define a similarity function for the comparison of two arbitrary events.

6.5.3 Measuring Event Similarity

For setting up a similarity function for spatio-temporal events, we use, in addition to event document profiles, the following event-related concepts that have been introduced in the Chapter 4:

- temporal / geographic mapping functions (Definition 4.8 and Definition 4.11, pages 138, 143)
- algorithms to map chronons / locations for equality (Algorithm 4.2, Algorithm 4.4; pages 141, 146)

The event similarity function compares two spatio-temporal events based on the semantics of their temporal and geographic components. Note, however, that while a spatio-temporal event is defined as $\langle t_i, s(t)_i, p(t)_i, g_j, s(g)_j, p(g)_j \rangle$, i.e., as a tuple of an extracted temporal expression and an extracted geographic expression, the similarity is determined solely based on the hierarchy-relevant temporal and geographic semantics of the events, which we refer to as chronon c and geographic value v in the following. Furthermore, in the context of determining event similarity, we call c and v the *dimensions* of an event $e = \langle c, v \rangle$.

Possible Types of Event Similarity

To be able to concisely define an event similarity function, we have to specify the requirements that should be satisfied by the similarity function. For this, we first list all possible similarity relationships that can hold between two events. Note that we do not distinguish the number of temporal and geographic mapping steps that have to be applied for one dimension to reach equality between the events, but use c^* for c and v^* for v being mapped to any coarser granularity. Given two events $e_1 = \langle c_1, v_1 \rangle$ and $e_2 = \langle c_2, v_2 \rangle$, the following similarity relationships can occur:

1. The values of both dimensions of the events are identical.

   (1.1) $c_1 = c_2$ and $v_1 = v_2$

2. The values of one dimension have to be mapped to a coarser granularity.

   (2.1) $c_1^* = c_2$ and $v_1 = v_2$ (2.3) $c_1 = c_2$ and $v_1^* = v_2$

   (2.2) $c_1^* = c_2^*$ and $v_1 = v_2$ (2.4) $c_1 = c_2$ and $v_1^* = v_2^*$

3. The values of both dimensions have to be mapped to a coarser granularity.
   (3.1) $c_1^* = c_2$ and $v_1^* = v_2$ (3.4) $c_1^* = c_2$ and $v_1^* = v_2^*$
   (3.2) $c_1^* = c_2$ and $v_1 = v_2^*$ (3.5) $c_1^* = c_2^*$ and $v_1^* = v_2^*$
   (3.3) $c_1^* = c_2^*$ and $v_1^* = v_2$
4. It is not possible to map the values of one dimension to a coarser granularity to achieve equality.
   (4.1) $c_1^* \neq c_2^*$ and $v_1 = v_2$ (4.4) $c_1 = c_2$ and $v_1^* \neq v_2^*$
   (4.2) $c_1^* \neq c_2^*$ and $v_1^* = v_2$ (4.5) $c_1^* = c_2$ and $v_1^* \neq v_2^*$
   (4.3) $c_1^* \neq c_2^*$ and $v_1^* = v_2^*$ (4.6) $c_1^* = c_2^*$ and $v_1^* \neq v_2^*$
5. It is not possible to map the values of both dimensions to a coarser granularity to achieve equality.
   (5.1) $c_1^* \neq c_2^*$ and $v_1^* \neq v_2^*$

Note that the similarity relationships (4.1) to (5.1) can only occur if the underlying temporal and geographic hierarchies do not have a single root element, e.g., on the hierarchy levels T_{global} and G_{global}. Limiting the coarsest temporal and geographic granularities can be done to determine when events are too dissimilar to speak about "similarity" at all.

Requirements for the Event Similarity Function

Based on the listed set of possible similarity relationships, we now formulate the requirements for the similarity function $sim_e(e_1, e_2)$ and give a detailed description to each requirement.

R1: The more similar e_1 and e_2, the higher $sim_e(e_1, e_2)$.

The more similar two events e_1 and e_2 are, the higher should be their similarity $sim_e(e_1, e_2)$, with $sim_e(e_1, e_2)$ being maximal if both events are identical. In (1.1), the events are identical and thus should have the highest possible similarity score for events in the respective timeline and granularity.

R2: The fewer values of the same dimension need to be mapped, the higher $sim_e(e_1, e_2)$.

In the second group, either at least one of the chronons has to be mapped to a coarser timeline (2.1 and 2.2) or at least one normalized geographic value has to be mapped to a coarser granularity (2.3 and 2.4). If only one value has to be mapped (2.1 and 2.3), the similarity score should be higher than if both values have to be mapped (2.2 and 2.4). This reflects the fact that events with the relationship (2.1) or (2.3) can be identical real-world events – although described in an imprecise way – while events with relationship (2.2) or (2.4) cannot be identical real-world events.

For example, the sentences *"he visits NYC in May 2010"* and *"he visits NYC on May 4, 2010"* can be about the identical event. In contrast, the sentences *"he visits NYC on May 4, 2010"* and *"he visits NYC on May 10, 2010"* cannot be about the same event. Nevertheless, there still is a similarity between e_1 and e_2 in the latter example, since both events happened at the same location and temporally close to each other (both in May 2010). Consequently, $sim_e(e_1, e_2)$ should be higher for (2.1) and (2.3) and penalize (2.2) and (2.4).

The same cases occur in the third group: If only one value of each dimension has to be mapped as in (3.1) and (3.2), the similarity score should be higher than if both values of the same dimension are involved as in (3.3), (3.4), and (3.5).

R3: If mapping leads to no equality, $sim_e(e_1, e_2)$ should be 0.

In the fourth group, either the chronons or the geographic values cannot be mapped to a coarser granularity to achieve equality. In the fifth group both types of normalized values cannot be mapped sufficiently. Again, note that whether such cases can occur depends on the used hierarchies for geographic and temporal mappings. For example, if "Earth" is used as the top level of the geographic hierarchy, then every geographic expression can be mapped to the top of the hierarchy. However, if "country" is at the top level, then, e.g., cities located in different countries cannot be mapped to a coarser granularity to achieve equality. Thus, if no sufficient mapping can be applied, the assigned similarity score $sim_e(e_1, e_2)$ equals 0, even though there may be a temporal or geographic similarity when using a different hierarchy. Nevertheless, such unmatched events influence the aggregated similarity score when comparing two documents, as will be detailed below.

R4: The fewer mapping steps are needed, the higher $sim_e(e_1, e_2)$.

The similarity score additionally depends on the differences of granularity between either c and c^* or v and v^*. The granularities are represented by the timelines for temporal information and by the containment hierarchy for geographic expressions. The larger the differences, the less precise the information, and thus the lower $sim_e(e_1, e_2)$.

R5: The finer the granularities, the higher $sim_e(e_1, e_2)$.

So far, the original granularities of the values, i.e., before they are mapped to coarser granularities, are not taken into account. For example, if there are two events $e_1 = \langle$(2006), (Germany)$\rangle$ and $e_2 = \langle$(2006-07-09), (Berlin,Germany)$\rangle$, then $sim_e(e_1, e_1)$ should not equal $sim_e(e_2, e_2)$, as the similarity score should be sensitive to the original granularities of the events in the documents. An event that is mentioned more fine-grained in the document should be weighted higher than a coarser one, i.e., $sim_e(e_1, e_1) < sim_e(e_2, e_2)$ should hold.

Event Similarity Function

We now formalize a function $sim_e(e_1, e_2)$ that satisfies these requirements. As stated above, $sim_e(e_1, e_2) > 0$ only holds if equality of two events $e_1 = \langle c_1, v_1 \rangle$ and $e_2 = \langle c_2, v_2 \rangle$ can be achieved, namely by applying a certain number of mapping steps to the geographic and temporal dimensions of the events.

We define $\alpha = \alpha_t + \alpha_g$ as the number of mapping steps that are needed to achieve equality for events e_1 and e_2 in both dimensions. Specifically, α_t is the sum of the number of temporal mapping steps that need to be applied to c_1 and c_2 in order to achieve equality in the temporal dimension. Accordingly, α_g is the corresponding sum of the number of mapping steps applied to v_1 and v_2 in the geographic dimension. That is, $\alpha \in \{0, \ldots, k+l\}$ with k being the total number of possible geographic and l the total number of possible temporal mapping steps. Furthermore, we define β to be the maximum of the number of values per dimension that are involved in the mapping, thus $\beta \in \{0, 1, 2\}$. Based on α and β, we tentatively define the event-centric similarity $sim_e(e_1, e_2)$ to be calculated in the following way:

$$sim_e^{tentative}(e_1, e_2) = \frac{1}{(1+\alpha)^\beta} \tag{6.1}$$

While α is used to moderately decrease $sim_e(e_1, e_2)$, β increases the denominator exponentially, thus penalizing the similarity score stronger than α. This is motivated by requirement R2 that e_1 and e_2 can refer to the same event if $\beta = 1$, but cannot if $\beta = 2$ – no matter how large α is.

Equation 6.1 satisfies requirements R1 through R4 as will be shown below. However, it does not yet support R5 – the finer the original granularities, the higher $sim_e(e_1, e_2)$. Thus, we additionally consider a parameter α_{poss}, which is the number of mapping steps (both temporal and geographic) that are still possible for e_1 and e_2 after both events have been mapped to be equal in both dimensions.

Assuming that the temporal and geographic granularities were $\mathcal{T} = \{T_{day}, T_{month}, T_{year}\}$ and $\mathcal{G} = \{G_{city}, G_{country}\}$, the following example shows how to calculate α_{poss}: If e_1 and e_2 were both mapped to $\langle$(2006-06), (Germany)$\rangle$, then $\alpha_{poss} = 1$, as no further mapping step is possible for v and one more mapping step is possible for c (i.e., to the year timeline). By weighting $sim_e(e_1, e_2)$ with $(\alpha_{poss} + 1)$, R5 is supported by our similarity function. Adding 1 to α_{poss} is necessary as the similarity of the coarsest granularity would be 0 otherwise.

$$sim_e(e_1, e_2) = \frac{1}{(1+\alpha)^\beta} \times (\alpha_{poss} + 1) \tag{6.2}$$

Equation 6.2 satisfies all requirements R1 through R5, and can thus be used for calculating the similarity of two events. To exemplarily verify this, we calculate the similarity scores between four events (Table 6.1) and show that all five requirements are met. For better readability, we demonstrate this example using only the timelines $\mathcal{T} = \{T_{day}, T_{month}, T_{year}\}$ and the geographic granularities $\mathcal{G} = \{G_{city}, G_{country}\}$ for the temporal and geographic dimensions, respectively.

Although R1 – the more similar e_1 and e_2, the higher $sim_e(e_1, e_2)$ – is a subjective formulation, there are some examples in Table 6.1 for which this formulation is obvious, e.g., e_4 is more similar to e_3 than to e_1. This shows that sim_e is calculated correctly with respect to R1, since $sim_e(e_3, e_4) > sim_e(e_1, e_4)$. R4 – the fewer mapping steps are needed, the higher sim_e – is considered by sim_e since, e.g., $sim_e(e_3, e_4)$ (one mapping step is needed) is higher than $sim_e(e_1, e_4)$ (three mapping steps are needed). The fact that R2 is taken into account can be shown directly using Equation 6.2. If zero, one, or two values of the same dimension need to be mapped, then β equals 0, 1, or 2, respectively. For $\beta = 0$, the denominator of Equation 6.2 equals 1. If $\beta > 0$, then $\alpha > 0$ and thus, $(1 + \alpha) < (1 + \alpha)^2$, i.e., R2 is satisfied since $sim_e(\beta = 1) > sim_e(\beta = 2)$ for identical α values. The consideration of R5 – the finer the granularities, the higher $sim_e(e_1, e_2)$ – is already achieved by the modification from Equation 6.1 to Equation 6.2. Finally, if no equality can be achieved, Equation 6.2 is defined as $sim_e(e_1, e_2) = 0$, i.e., R4 is satisfied.

6.5.4 Event-centric Document Similarity Model

Defining the similarity of just two events is already not trivial and many requirements need to be satisfied by the similarity function, as discussed above. However, aggregating the similarity of two sets of events in a meaningful way is even more challenging. Therefore, before defining how to calculate a respective aggregation, we first define some requirements for this aggregation. Then, the event-centric document similarity model satisfying these requirements is incrementally developed.

(a) Event examples.

id	event
e_1	⟨(2006),(Germany)⟩
e_2	⟨(2006-07),(Stuttgart,Germany)⟩
e_3	⟨(2006-06),(Berlin,Germany)⟩
e_4	⟨(2006-06-09),(Berlin,Germany)⟩

(b) Event similarity scores.

	e_1	e_2	e_3	e_4
e_1	1	0.33	0.33	0.25
e_2		3	0.04	0.03
e_3			3	1.5
e_4				4

Table 6.1: Event examples (a) and similarity scores between them calculated using Equation 6.2 (b).

Requirements for the Event-centric Document Similarity Model

To be able to specify a suitable similarity function, we first formalize some requirements for the aggregation of event similarity scores that need to be satisfied. As already pointed out, we define two documents to be similar, the more similar the events in the documents are.

A1: The more matching events are in d_1 and d_2, the higher $d\text{-}sim_e(d_1, d_2)$.

A2: The more non-matching events are in d_1 and d_2, the more $d\text{-}sim_e(d_1, d_2)$ should be penalized.

A3: If only one document contains additional events, this should not be penalized as much as if both documents contain additional non-matching events.

In addition, all the requirements formulated for event similarity apply here, too. That is, requirements R1 to R5 described in the previous section can be summarized as:

A4: The more similar the events in d_1 and d_2, the higher $d\text{-}sim_e(d_1, d_2)$.

Aggregation of Event Similarity Scores

Given a document, the objective now is to create a ranked list of most similar documents using the information given by their event document profiles. The simplest way to calculate this similarity is to view all events as terms. For every document, these terms then form a vector so that the similarity between two documents can be calculated by comparing their vectors with, e.g., the cosine similarity function. This simple approach satisfies A1. However, other requirements are not satisfied, in particular A4 is not taken into account at all because only identical events can be considered.

The vector approach is thus not applicable since we do not want to consider only exact matches of events but also similar events after granularity mapping. Therefore, instead of comparing vectors, we perform event alignment by building the cross-product of the event document profiles to compare all event pairs. If two events are not equal, we apply Algorithm 4.2 and Algorithm 4.4 to map chronons and locations for equality, respectively. Thus, they will be mapped to coarser granularities until equality is reached or no further mapping is possible. The similarity score for every pair is calculated according to $sim_e(e_1, e_2)$ (cf. Equation 6.2) and aggregated to $d\text{-}sim_e(d_1, d_2)$.

$$d\text{-}sim_e(d_1, d_2)^{tentative} = \sum_{i=0}^{n} \sum_{j=0}^{m} sim_e(e_i, e_j) \tag{6.3}$$

However, requirement A2 is not satisfied so far. Therefore, we have to normalize $d\text{-}sim_e(d_1, d_2)$ according to the number of events in the documents. For two documents d_1 and d_2 containing n and m events, respectively, using the sum $n + m$ violates A3. Thus, we use $min(n, m)$ for normalization and $d\text{-}sim_e(d_1, d_2)$ is thus calculated as follows:

$$d\text{-}sim_e(d_1, d_2) = \frac{\sum_{i=0}^{n} \sum_{j=0}^{m} sim_e(e_i, e_j)}{min(n, m)} \tag{6.4}$$

This equation for calculating event-centric document similarity satisfies the requirements A1 to A4, and we will refer to this event-centric similarity model as "full model" (FM). However, we can also divide this model into its three main features that will also be analyzed in the evaluation.

- *granularity mapping (M)* : When comparing two different events, it is first checked if the two events can be mapped to equality, and, if it is possible, both dimensions of the events are mapped to coarser granularities until equality is reached. During this step, α and β are determined and the event similarity is calculated as in Equation 6.1. If this feature is not considered, the event similarity between different events is always 0.
- *granularity weighting (W)* : This feature takes care of the granularities of matching events, i.e., of requirement R5 for event similarity. Thus, considering the granularity weighting factor, the event similarity scores are calculated according to Equation 6.2.
- *event quantity normalization (N)* : The aggregated similarity score calculated for two documents is normalized with respect to the number of events in their event document profiles. Thus, if the feature is considered, $d\text{-}sim(d_1, d_2)$ is calculated according to Equation 6.4, and, if the feature is not considered, it is calculated according to Equation 6.3.

Considering the features M, W, and N and thus applying Equation 6.4 for event-centric document similarity with Equation 6.2 for calculating event similarity scores, the requirements A1 to A4 are satisfied. In the evaluation, we refer to this document similarity model as full model (FM). To study the influence of the single features, we will also disregard single features and all combinations of them. The notation for the modified models will be "FM -M" (no mapping), "FM -W" (no weighting), "FM -N" (no normalization), ..., "FM -WN" (no weighting and no normalization) and "FM -MWN" (no mapping, no weighting, no normalization). The evaluation setting will be described in Section 6.5.6. However, we first explain how to compute event-centric document similarity.

6.5.5 Similarity Calculation

To calculate $d\text{-}sim_e(d_1, d_2)$, the event similarity scores for the cross-product of all events in the event document profiles of the two documents have to be computed. In Algorithm 6.1, the procedure to determine event similarity is presented. After mapping the chronons and locations to equality (lines 3 and 4) using the Map_Chronons_For_Equality and Map_Locations_For_Equality procedures (Algorithm 4.2, page 141 and Algorithm 4.4, page 146), it is checked whether or not these mappings were successful (line 5). If not, the similarity equals 0 (line 17). Otherwise, α equals the sum of the temporal and geographic mapping steps for the two events (line 6), and β is determined (lines 7 to 12). Finally, the event similarity is calculated based on the remaining possible mapping steps (line 14), which depend on the underlying timeline and granularity hierarchies, and the similarity score is returned (line 15).

Algorithm 6.1 Procedure to calculate event similarity for two events e_1 and e_2.

```
1:  procedure CALCULATE_SIM_E(e1, e2)
2:      sim = 0
3:      αt1, αt2, timeline = MAP_CHRONONS_FOR_EQUALITY(e1.t, e2.t)
4:      αg1, αg2, granularity = MAP_LOCATIONS_FOR_EQUALITY(e1.g, e2.g)
5:      if NOT ((timeline = null) or (granularity = null)) then
6:          α = αt1 + αt2 + αg1 + αg2
7:          β = 1
8:          if (α = 0) then
9:              β = 0
10:         else if ((αt1 > 0) and (αt2 > 0)) or ((αg1 > 0) and (αg2 > 0)) then
11:             β = 2
12:         end if
13:         poss = αposs(timeline, granularity)
14:         sim = 1/(1+α)^β * (poss + 1)
15:         return sim
16:     end if
17:     return 0
18: end procedure
```

In Algorithm 6.2, the procedure to calculate event-centric document similarity for two documents is presented. The event similarity is calculated for the cross product of all events in the event document profiles of the documents (lines 3 and 4) by summing up the results of the CALCULATE_SIM_E procedure for each event pair (line 5). The similarity score is then length-normalized (line 8) and returned (line 9).

An example how to calculate event-centric document similarity is presented in Table 6.2. At the top of Table 6.2(a)-(c), the original events of the three example documents d_1, d_2, and d_3 are depicted, respectively. Below the original events, the original events and all their possible mappings are listed and grouped by α_{poss}. Note that for better clarity and readability, we again only use the timelines $\mathcal{T} = \{T_{day}, T_{month}, T_{year}\}$ and the geographic granularities $\mathcal{G} = \{G_{city}, G_{country}\}$ as temporal and geographic concepts in the hierarchies, respectively. In Table 6.2(d)-(f), we show how the event-centric document similarity scores between d_1, d_2, and d_3 are calculated by first listing the similarity scores of single event pairs and then depicting the aggregated scores.

In summary, using the above approach, the event-centric document similarity measure can be computed solely based on document event profiles. Obviously, the algorithm to compute event-centric document similarity can easily be optimized. For instance, each similarity between two events could be stored. Then, independent of in how many documents such an event pair occurs, the similarity score have to be computed only once. In addition, if the highest temporal and geographic hierarchy levels are not "Earth" and "anytime", i.e., if it is possible that two events cannot be mapped to equality, it can be validated if a temporal and a geographic mapping to equality is possible, before trying to calculate similarity scores.

Although we apply several optimizations in our implementation of the algorithm, we do not further detail them here because documents usually do not contain huge amounts of events and thus the computational overhead is rather limited. Thus, we now present the evaluation objectives and setup as well as the evaluation corpora and evaluation results in the next sections.

(a) Example document d_1.

original events		
$\alpha_{poss} = 3$	a	⟨(2006-07-09),(Berlin, Germany)⟩
$\alpha_{poss} = 3$	b	⟨(2006-06-09),(Munich, Germany)⟩
$\alpha_{poss} = 1$	c	⟨(2006-06),(Germany)⟩
$\alpha_{poss} = 0$	d	⟨(2006),(Germany)⟩
mappings of events		
$\underline{\alpha_{poss} = 3}$	$a_{0,0}$	⟨(2006-07-09),(Berlin, Germany)⟩
	$b_{0,0}$	⟨(2006-06-09),(Munich, Germany)⟩
$\underline{\alpha_{poss} = 2}$	$a_{1,0}$	⟨(2006-07),(Berlin, Germany)⟩
	$b_{1,0}$	⟨(2006-06),(Munich, Germany)⟩
	$a_{0,1}$	⟨(2006-07-09),(Germany)⟩
	$b_{0,1}$	⟨(2006-06-09),(Germany)⟩
$\underline{\alpha_{poss} = 1}$	$a_{2,0}$	⟨(2006),(Berlin, Germany)⟩
	$b_{2,0}$	⟨(2006),(Munich, Germany)⟩
	$a_{1,1}$	⟨(2006-07),(Germany)⟩
	$b_{1,1}$, $c_{0,0}$	⟨(2006-06),(Germany)⟩
$\underline{\alpha_{poss} = 0}$	$a_{2,1}$, $b_{2,1}$, $c_{1,0}$, $d_{0,0}$	⟨(2006),(Germany))⟩

(b) Example document d_2.

original events		
$\alpha_{poss} = 3$	e	⟨(2006-07-08),(Bonn, Germany)⟩
$\alpha_{poss} = 3$	f	⟨(2006-07-09),(Berlin, Germany)⟩
$\alpha_{poss} = 1$	g	⟨(2006),(Germany)⟩
mappings of events		
$\underline{\alpha_{poss} = 3}$	$e_{0,0}$	⟨(2006-07-08),(Bonn, Germany)⟩
	$f_{0,0}$	⟨(2006-07-09),(Berlin, Germany)⟩
$\underline{\alpha_{poss} = 2}$	$e_{1,0}$	⟨(2006-07),(Bonn, Germany)⟩
	$f_{1,0}$	⟨(2006-07),(Berlin, Germany)⟩
	$e_{0,1}$	⟨(2006-07-08),(Germany)⟩
	$f_{0,1}$	⟨(2006-07-09),(Germany)⟩
$\underline{\alpha_{poss} = 1}$	$e_{2,0}$	⟨(2006),(Bonn, Germany)⟩
	$f_{2,0}$	⟨(2006),(Berlin, Germany)⟩
	$e_{1,1}$, $f_{1,1}$	⟨(2006-07),(Germany)⟩
$\underline{\alpha_{poss} = 0}$	$e_{2,1}$, $f_{2,1}$, $g_{0,0}$	⟨(2006),(Germany)⟩

(c) Example document d_3.

original events		
$\alpha_{poss} = 3$	h	⟨(2006-07-08),(Bonn, Germany)⟩
mappings of events		
$\underline{\alpha_{poss} = 3}$	$h_{0,0}$	⟨(2006-07-08),(Bonn, Germany)⟩
$\underline{\alpha_{poss} = 2}$	$h_{1,0}$	⟨(2006-07),(Bonn, Germany)⟩
	$h_{0,1}$	⟨(2006-07-08),(Germany)⟩
$\underline{\alpha_{poss} = 1}$	$h_{2,0}$	⟨(2006),(Bonn, Germany)⟩
	$h_{1,1}$	⟨(2006-07),(Germany)⟩
$\underline{\alpha_{poss} = 0}$	$h_{2,1}$	⟨(2006),(Germany)⟩

(d) Similarity calculation for d_1 and d_2.

pair	match	α	β	α_{poss}	sim_e
a, e	$a_{1,1}$, $e_{1,1}$	4	2	1	0.08
a, f	$a_{0,0}$, $f_{0,0}$	0	0	3	4
a, g	$a_{2,1}$, $g_{0,0}$	3	1	0	0.25
b, e	$b_{2,1}$, $e_{2,1}$	6	2	0	0.02
b, f	$b_{2,1}$, $f_{2,1}$	6	2	0	0.02
b, g	$b_{2,1}$, $g_{0,0}$	3	1	0	0.25
c, e	$c_{1,0}$, $e_{2,1}$	4	2	0	0.04
c, f	$c_{1,0}$, $f_{2,1}$	4	2	0	0.04
c, g	$c_{1,0}$, $g_{0,0}$	1	1	0	0.5
d, e	$d_{0,0}$, $e_{2,1}$	3	1	0	0.25
d, f	$d_{0,0}$, $f_{2,1}$	3	1	0	0.25
d, g	$d_{0,0}$, $g_{0,0}$	0	0	0	1.0
$d\text{-}sim(d_1, d_2) = \frac{\sum_i \sum_j sim_e(e_i, e_j)}{min(n,m)} = 2.23$					

(e) Similarity calculation for d_1 and d_3.

pair	match	α	β	α_{poss}	sim_e
a, h	$a_{1,1}$, $h_{1,1}$	4	2	1	0.08
b, h	$b_{2,1}$, $h_{2,1}$	6	2	0	0.02
c, h	$c_{1,0}$, $h_{2,1}$	4	2	0	0.04
d, h	$d_{0,0}$, $h_{2,1}$	3	1	0	0.25
$d\text{-}sim(d_1, d_3) = \frac{\sum_i \sum_j sim_e(e_i, e_j)}{min(n,m)} = 0.39$					

(f) Similarity calculation for d_2 and d_3.

pair	match	α	β	α_{poss}	sim_e
e, h	$e_{0,0}$, $h_{0,0}$	0	0	3	4
f, h	$f_{1,1}$, $h_{1,1}$	4	2	1	0.08
g, h	$g_{0,0}$, $h_{2,1}$	3	1	0	0.25
$d\text{-}sim(d_2, d_3) = \frac{\sum_i \sum_j sim_e(e_i, e_j)}{min(n,m)} = 4.33$					

Table 6.2: Calculating event-centric document similarity scores between three example documents. Original events and their mappings contained in d_1 (a), d_2 (b), and d_3 (c). Indices of event ids represent α_t and α_g. Similarity calculations are show in (d), (e), and (f).

Algorithm 6.2 Procedure to calculate event-centric document similarity for two documents d_1 and d_2 based on their event document profiles ($edp1$ and $edp2$).

```
1: procedure CALCULATE_D_SIM_E(edp1, edp2)
2:     sim = 0
3:     for all e1 in edp1 do
4:         for all e2 in edp2 do
5:             sim = sim+ CALCULATE_SIM_E(e1, e2)
6:         end for
7:     end for
8:     sim = sim/min(edp1.length, edp2.length)
9:     return sim
10: end procedure
```

6.5.6 Evaluation Scenarios

Evaluating event-centric document similarity is a challenging task. In addition to the general subjectivity issue when dealing with similarity concepts, no adequate gold standard is available. We cannot use standard similarity evaluation corpora as our goal is not to identify documents as similar that talk about the same topic in general, but only documents that contain similar events. Although there are evaluation corpora for related tasks such as topic detection and tracking (TDT), these are not suitable due to the different goals of TDT and our similarity model. While TDT systems associate a main event with documents and cluster incoming news articles according to these events, we take into account all events extracted from documents to calculate event-centric similarity scores.

Manual Evaluation

A straightforward way to evaluate our model is thus to select a corpus, calculate similarity scores for all document pairs and manually check whether two documents are similar from an event-centric perspective. However, this scenario is very labor-intensive and can thus only be done for a small subset of documents.

Cross-language Evaluation

Another way to evaluate our model is based on a multilingual corpus containing cross-language links between related documents from different languages. Intuitively, documents written in different languages having the same major topic (e.g., about the same person) can be assumed to be similar in an event-centric way. For example, the English and the German versions of a biography can obviously be regarded as similar with respect to the mentioned events (e.g., birth, death, travels) – no matter whether or not the two documents are (partial) translations of each other. Note that it is important that the documents are documents written in different languages and not documents about the same topic in general because otherwise document similarity could also be detected due to similar word occurrences independent of occurring events.

Using a multilingual corpus containing cross-language links, we can evaluate how often cross-language linked documents are the top-k most similar documents for each other. Of course, the cross-language links are only used for evaluation purposes, and not considered for calculating the similarity scores. This second evaluation scenario allows for a large-scale evaluation.

Note that it is not necessary that language-linked documents are translations of each other, i.e., there is no need for a parallel corpus. Although parallel corpora have been used to evaluate cross-language similarity models, e.g., Steinberger et al. "measur[ed] the number of times the translation of a given document was identified as the most similar document" (Steinberger et al., 2002) by their multilingual thesaurus-based similarity model, it is rather obvious that verbatim translations will be determined as very similar with our event-centric model. Assuming that high quality temporal tagger and geo-tagger are used, only identical events will be extracted from such document pairs. Thus, to increase the difficulty and to achieve more meaningful results, we will not make use of a parallel corpus but of language-linked Wikipedia articles as will be detailed in the next section.

Comparison to a Term-based Approach

Finally, a further evaluation experiment is set up to analyze whether our event-centric similarity model finds other types of similarity than term-based approaches. As already explained, we expect to find other kinds of documents to be similar compared to term-based methods, i.e., we do not aim at improving other similarity measures but want to show that other kinds of similarity are detected. Thus, for comparison with term-based models, we do not have to use highly sophisticated methods such as latent semantic analysis, but we can use a simple model as representative for term-based approaches. For this, we select the tf-idf measure combined with the cosine similarity.

Event Extraction

For extracting spatio-temporal events from the documents, we will apply the cooccurrence approach with the simple sub-sentence feature. As described in Section 4.5, other extraction methods can be used to improve the precision in the event extraction process. However, using the cooccurrence approach has the following advantages: (i) some of the more sophisticated methods are not fully language-independent, and (ii) a high recall in the event extraction process is important to be able to detect as many similarity relations between documents as possible. In addition, to demonstrate the effectiveness of the model, no highly sophisticated event extraction method is necessary, but, of course, the cooccurrence approach could be replaced by any other event extraction method.

Temporal and Geographic Hierarchies

As mentioned above, it is either possible to use T_{global} and G_{global} as highest temporal and geographic hierarchies for the similarity calculation or to use lower levels in the hierarchies as root elements. Since events matching only on T_{global} or G_{global} can hardly be considered as similar, we make use of the following temporal and geographic hierarchies in our experiments: $\mathcal{T} = \{T_{time}, T_{day}, T_{month}, T_{year}, T_{decade}\}$ and $\mathcal{G} = \{G_{POI}, G_{city}, G_{state}, G_{country}\}$.

Summary

In summary, we use the cooccurrence approach for event extraction, limit the temporal and geographic hierarchies to T_{decade} and $G_{country}$, respectively, and apply three types of evaluations. The scenarios together with the respective evaluation objective can be summarized as follows:

- **Manual evaluation** on a small set of documents to demonstrate that detected event-centric similarity relations are meaningful.

(a) Documents per language.

English (en)	4,321
German (de)	2,491
French (fr)	3,256
Spanish (sp)	2,767
total documents	12,835

(b) Language distribution.

4 languages	2,097
3 languages	772
2 languages	679
English only	773
distinct documents	4,321

(c) Document pairs per language pair.

en, de	2,491	de, fr	2,368
en, fr	3,256	de, sp	2,137
en, sp	2,767	fr, sp	2,558
total document pairs		15,577	

Table 6.3: Language statistics of the documents in the FA-4lang corpus, showing the number of documents per language (a), in how many languages the documents are available (b), and how many language-linked document pairs are available per language pair (c).

- **Cross-language evaluation** on a large, multilingual corpus to demonstrate that detected event-centric similarity relations are meaningful.
- **Comparison to term-based approach** to demonstrate that other types of similarity are detected.

In the next section, we describe the used data sets and in particular explain the creation of our multilingual corpus.

6.5.7 Evaluation Corpora

For our experiments, we use two corpora: (i) a multilingual Wikipedia featured articles corpus containing the English featured articles and their language-linked articles in German, French, and Spanish, and (ii) a subset of the Wikipedia XML corpus (Denoyer and Gallinari, 2006), namely all documents that are available in English and German. While the first corpus contains less but longer documents, the second corpus contains more but shorter documents.

Multilingual Wikipedia Featured Articles Corpus

For creating the multilingual Wikipedia featured articles corpus (FA-4lang corpus), we crawled all English Wikipedia featured articles,[3] and – if available – the German, French, and Spanish articles linked to the English ones through a cross-language link.[4] In Table 6.3, we show some statistics of the corpus.

As presented in Table 6.3(a), there are 4,321 English articles, and the corpus contains 12,835 documents in total. Note that not every English article also exists in other languages. As shown in Table 6.3(b), some articles are available in all four languages, but others only in three or two languages, and there are even some articles that exist only in English, i.e., some documents in the corpus do not build a cross-language pair with any other document. This language distribution of the articles will be important in the evaluation described below. Table 6.3(c) shows the number of document pairs for each language pair – information that we will also refer to below when describing the evaluation results.

[3] `http://en.wikipedia.org/wiki/Wikipedia:Featured_articles` [last accessed August 3, 2014].

[4] In the paper in which we introduced the event-centric document similarity model (Strötgen et al., 2011), we also performed experiments on a multilingual Wikipedia featured articles corpus. However, due to the lack of temporal taggers for other languages, we only used German and English articles. Applying HeidelTime for temporal tagging, we can now extract spatio-temporal events in more languages and performed new experiments on a multilingual corpus consisting of documents in four languages instead of only two. For evaluation result on the initial corpus, we refer to Strötgen et al. (2011).

The reasons for choosing the Wikipedia featured articles for building our multilingual evaluation corpus are that (i) they are determined by the editors to be of high quality, and (ii) they are grouped into categories and biography subcategories. Note that in general, several categories are associated with Wikipedia articles but we use the featured article categories for having a single main category assigned to each document. This category information allows for a detailed analysis which documents contain many events and for which topics our similarity model is particularly suitable.

As already motivated in Chapter 4, spatio-temporal events are particularly frequent in documents about, e.g., persons or history, while they hardly occur in other document types. Obviously, if a document does not contain any events, no similarity scores can be calculated for this document and any other document with our event-centric model. In Table 6.4, we present details about the FA-4lang corpus grouped by category and language, such as the number of documents in total, the number of documents without any events, the average count of events per document, and the number of language-linked document pairs.

As shown in Table 6.4, the average number of events per document is much higher for English than for the three other languages independent of the category. This can be explained by the fact that Wikipedia's featured articles tend to be quite long and that most of the language-linked documents in German, French, and Spanish are not featured and thus also much shorter than the English ones.

Furthermore, it can be observed that the percentage of document pairs for which both documents contain at least one spatio-temporal event is very high for documents about history, biographies, and wars, while it is rather low for documents about mathematics and computing, for instance. Obviously, in the latter categories spatio-temporal events do not play an important role. Thus, we expect that our event-centric document similarity model performs much better for documents of the upper listed categories in Table 6.4.

Wikipedia XML Corpus

As a second corpus, we aimed at a larger multilingual corpus to evaluate our model with even more documents taken into account. For this, we use the publicly available Wikipedia XML Corpus (Denoyer and Gallinari, 2006), containing Wikipedia articles as XML files. We selected the main collections of English and German articles consisting of 659,388 and 305,099 articles, respectively, and created a subset of all document pairs for which the English and the German articles are available, resulting in 94,348 document pairs.

In Table 6.5, we show some details of the Wiki-XML corpus[5] and of the FA-4lang corpus to allow for an easy comparison. In contrast to the documents of the FA-4lang corpus, each document of the Wiki-XML corpus builds a language-linked document pair with exactly one other document. Note that the large differences in the number of average events per document can mainly be explained by the different lengths of the documents. In addition, the Wiki-XML corpus contains several very short documents with just a couple of sentences so that the number of document pairs for which both documents contain at least one event is much smaller than the total number of document pairs. However, with 46,201 document pairs, there are still twice as many document pairs containing events than in the FA-4lang corpus.

[5] Although we already used the Wiki-XML corpus in our initial experiments in (Strötgen et al., 2011), we rerun all experiments using the latest HeidelTime version for temporal tagging. In addition to the fact that HeidelTime was significantly improved since its initial version, another reason is to allow for a better comparability between the results on the two corpora – which are processed in exactly the same way since we used the same components for the event extraction process.

category	English	German	French	Spanish	avg. events en	de	fr	sp	pairs	%
History	135 (0)	68 (4)	98 (5)	103 (2)	98	22	51	44	483 (25)	94.8
Biographies	1,058 (0)	722 (39)	819 (37)	647 (22)	82	20	39	33	4,052 (237)	94.2
Wars, battles, ev.	172 (0)	91 (6)	154 (14)	112 (5)	77	14	66	32	625 (51)	91.8
Culture, society	74 (0)	39 (4)	48 (6)	46 (1)	58	13	24	32	234 (27)	88.5
Sport, recreation	187 (0)	81 (10)	120 (16)	94 (4)	91	133	73	50	538 (68)	87.4
Awards, decorat.	25 (0)	21 (6)	25 (2)	21 (0)	63	7	27	28	130 (21)	83.8
Education	40 (0)	15 (2)	21 (2)	16 (3)	73	10	15	11	94 (16)	83.0
Art, architecture	133 (1)	70 (11)	111 (21)	92 (9)	53	8	21	21	496 (95)	80.8
Warfare	212 (0)	150 (25)	171 (30)	140 (15)	100	15	21	23	858 (167)	80.5
Royalty, nobility	8 (0)	4 (2)	3 (0)	3 (0)	67	19	46	52	19 (4)	78.9
Business, econo.	64 (0)	18 (2)	28 (9)	14 (1)	47	48	11	17	90 (20)	77.8
Politics, govern.	47 (0)	14 (2)	20 (4)	16 (3)	60	10	34	17	89 (20)	77.5
Law	49 (0)	10 (4)	24 (4)	17 (1)	44	7	34	13	83 (19)	77.1
Geography, plac.	210 (0)	131 (21)	173 (46)	125 (23)	110	25	46	41	795 (186)	76.6
Music	200 (1)	123 (35)	159 (45)	160 (18)	32	7	9	14	822 (234)	71.5
Meteorology	145 (1)	41 (8)	88 (29)	71 (10)	46	13	13	19	319 (91)	71.5
Literature, theat.	162 (2)	52 (11)	90 (30)	99 (13)	29	12	9	14	415 (122)	70.6
Engineering, tec.	43 (4)	25 (4)	33 (10)	26 (7)	40	8	15	10	149 (44)	70.5
Geology, geophy.	22 (2)	9 (4)	18 (3)	12 (2)	21	4	11	9	65 (20)	69.2
Health, medicine	50 (3)	40 (12)	44 (8)	44 (13)	27	9	13	10	247 (77)	68.8
Language, lingui.	12 (1)	7 (3)	10 (3)	9 (2)	43	9	12	23	47 (17)	63.8
Chemistry, mine.	37 (5)	34 (13)	35 (12)	35 (8)	20	2	9	6	203 (76)	62.6
Transport	163 (0)	59 (17)	73 (35)	43 (11)	72	12	16	14	292 (110)	62.3
Media	256 (10)	107 (31)	158 (55)	155 (47)	20	6	7	7	754 (291)	61.4
Video gaming	171 (17)	99 (29)	157 (63)	139 (40)	15	5	7	8	724 (285)	60.6
Philosophy, psy.	12 (0)	9 (4)	10 (3)	10 (3)	20	3	8	8	57 (23)	59.6
Religion, mystic.	44 (2)	21 (9)	37 (12)	33 (10)	41	10	29	19	162 (68)	58.0
Physics, astrono.	112 (10)	99 (49)	110 (27)	106 (37)	15	2	7	6	617 (286)	53.6
Biology	436 (4)	305 (140)	383 (132)	346 (144)	23	4	11	9	1,937 (914)	52.8
Food, drink	17 (0)	7 (4)	13 (3)	10 (5)	29	5	9	11	52 (28)	46.2
Computing	16 (1)	12 (7)	14 (8)	14 (3)	11	1	4	5	78 (45)	42.3
Mathematics	9 (0)	8 (6)	9 (5)	9 (4)	12	1	2	2	51 (35)	31.4
total	4,321 (64)	2,491 (524)	3,256 (679)	2,767 (466)	59	15	25	20	15,577 (3,722)	76.1

Table 6.4: FA-4lang corpus statistics per category showing the number of documents and average number of events per document separated by language; in parentheses the counts of documents without spatio-temporal events. In addition, the number of language-linked document pairs are reported with the counts of pairs for which not both documents contain at least one event. Categories are ordered by the percentage of document pairs for which both documents contain events.

	FA-4lang corpus					Wiki-XML corpus		
	English	German	French	Spanish	pairs	English	German	pairs
document (total)	4,321	2,491	3,256	2,767	15,577	94,384	94,384	94,384
documents (w. events)	4,257	1,967	2,577	2,301	11,855	64,396	52,439	46,201
average events	59	15	25	20		13	5	

Table 6.5: Comparing details such as the numbers of document pairs containing events for the two evaluation corpora FA-4lang corpus and Wiki-XML corpus.

6.5.8 Evaluation Results

In this section, we present the evaluation results for the different experiments. After describing the results of the cross-language experiments, we compare the detected event-centric similarity relations with standard term-based document similarity relations to show the differences, and finally present the results of the manual evaluation.

Cross-language Experiments

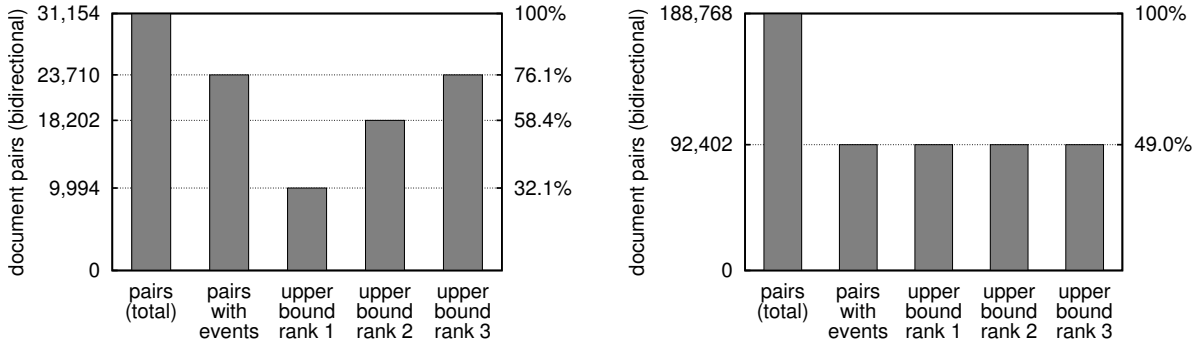
In the cross-language evaluation, we determine for each document containing at least one spatio-temporal event, the rank of its cross-language linked documents, i.e., each language-linked document pair is subject of analysis. Recall that we assume that cross-language linked documents are quite similar to each other with respect to events mentioned in the documents, although there might also be other documents that may be even more similar, e.g., if one document is quite long in one language and quite short in another language, there might be another long document containing many similar or identical events as the long document. However, in general, we expect cross-language linked documents to be quite similar – in particular for documents of categories such as history and biographies.

Note that given a document d_i, a cross-language linked document d_i' can only be determined as a similar document, if both d_i and d_i' contain at least one spatio-temporal event. In addition, if a document d_i has more than one cross-language linked document – as it often occurs in the FA-4lang corpus – only one of the linked documents can be determined as the most similar document. Thus, there is an upper bound for document pairs to be ranked as most similar documents for each other when using the FA-4lang corpus. Similarly, since some documents are available in four languages, there is also an upper bound for document pairs being ranked as second most similar ranked documents.

In Figure 6.4(a), we show the relations between document pairs in total, document pairs for which both documents contain at least one event, and the rank 1, rank 2, and rank 3 upper bounds. Similarly, Figure 6.4(b) shows those relations for the Wiki-XML corpus. Note that the rank 1, rank 2, and rank 3 upper bounds of the Wiki-XML corpus are identical to the number of document pairs for which both documents contain at least one event because this corpus contains only documents in two languages. Thus, the only prerequisite for an event-centric document similarity score is that both documents of a document pair contain at least one event.

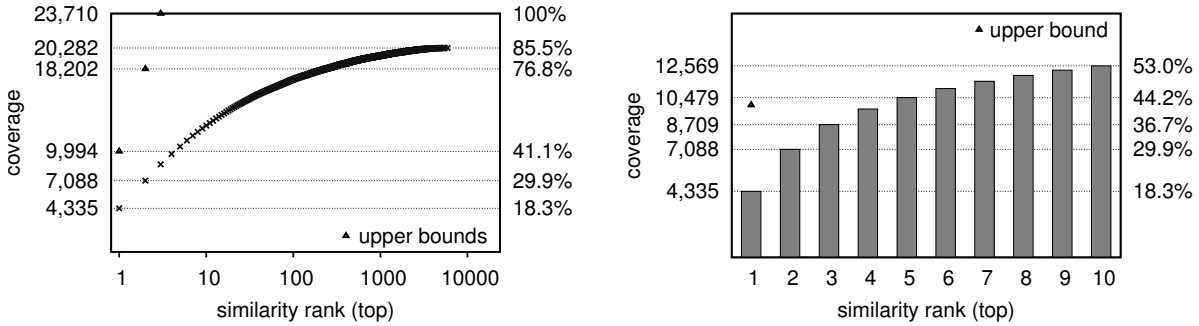
Cross-language Experiments – Full Model

In the following, we present the evaluation results of our event-centric document similarity model (full model, FM) on the FA-4lang corpus and the Wiki-XML corpus. Then, we analyze the influence of the three



(a) FA-4lang corpus. (b) Wiki XML corpus.

Figure 6.4: Document pair statistics on the FA-4lang corpus (a) and the Wiki-XML corpus (b). For all document pairs with events, it is possible to calculate an event-centric similarity score. The upper bounds are the maximum number of document pairs which can theoretically be determined to be rank 1, 2, or 3.

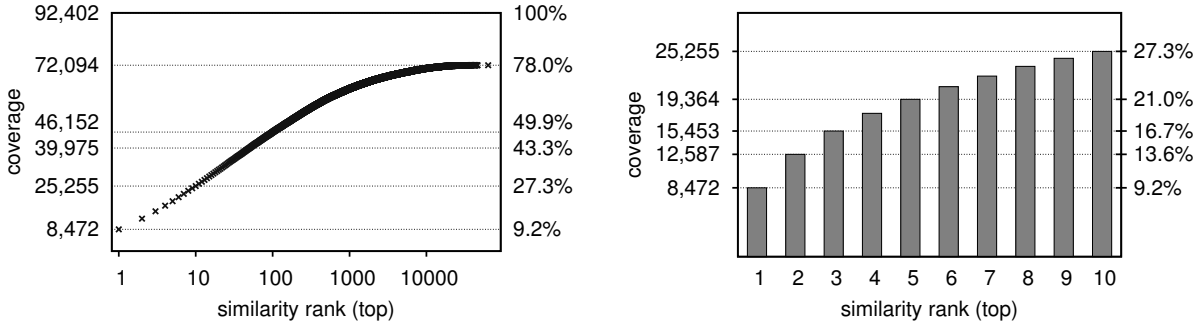


(a) Log-scale plot. (b) Top 10 ranks in detail.

Figure 6.5: Evaluation results of the full event-centric similarity model (FM) on the FA-4lang corpus for all document pairs containing events.

main features of the similarity model, i.e., granularity mapping (M), granularity weighting (W), and event quantity normalization (N) (cf. Section 6.5.4, page 230). While we first use the whole FA-4lang corpus, we then study further details by analyzing the results for all document categories. For the evaluation, we determine for each language-linked document pair (d_i, d_j), the similarity rank of d_i given d_j and vice versa.

For the FA-4lang corpus, the evaluation results of the full model are shown in Figure 6.5. In Figure 6.5(a), the similarity ranks are depicted for all cross-language linked documents. Note that all document pairs for which not both documents contain at least one event are excluded so that the 23,710 document pairs for which both documents contain events are set to 100%. A first result is that for 14.5% of the language-linked document pairs, no similarity is detected. The reasons for this are that we use T_{decade} and $G_{country}$ as coarsest temporal and geographic hierarchies so that the similarity between two very dissimilar events equals zero, and that the 14.5% of the document pairs obviously contain only such events. Note, however, that the results are on the full corpus, i.e., on documents of all types of categories. We will study below



(a) Log-scale plot. (b) Top 10 ranks in detail.

Figure 6.6: Evaluation results of the full event-centric similarity model (FM) on the Wiki-XML corpus for all document pairs containing events.

the performance differences on single categories. Despite using the full corpus, more than 18% of all document pairs are detected as the most similar documents for each other. Since the upper bound for rank 1 is at 41.1%, this can be considered as quite high.

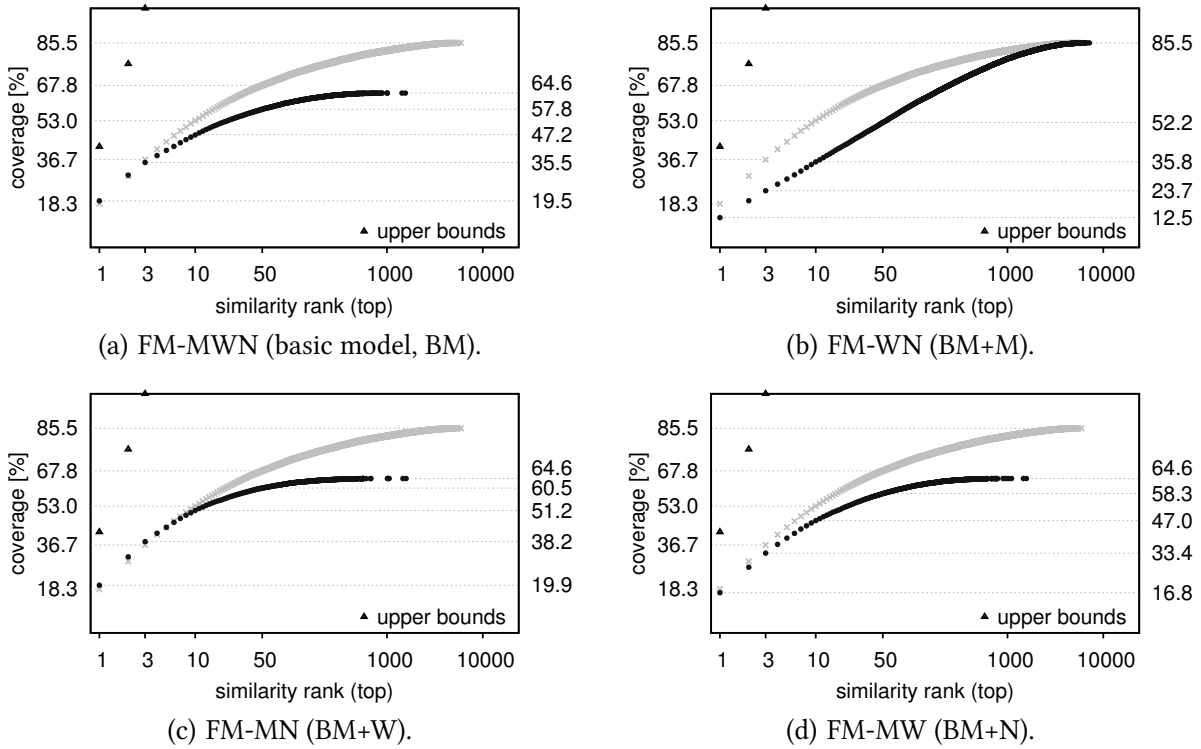
To get a better overview of the top ranks, we separately plot the evaluation results of the full model for ranks 1 to 10 in Figure 6.5(b). As can be seen, in almost 30% of all cases, a language-linked document is ranked as most or second most similar. This number increases to 53% when considering the top 10 ranks.

Similar to the results of the FA-4lang corpus, we show the results on the Wiki-XML corpus in Figure 6.6 for all document pairs containing events (a) and the top 10 results in detail (b). Note that due to the larger number of documents in the corpus, the chance for a cross-language linked document to be ranked among the most similar documents decreases. In addition, there may be more documents in the corpus being very similar to a query document in an event-centric way. This explains the lower results compared to the smaller FA-4lang corpus. Thus, with respect to the number of documents in the corpus, the results on the Wiki-XML corpus can still be considered as quite high. In 16.7%, a language-linked document is considered as one of the three most similar documents for its linked document. Considering documents that are ranked as one of the ten most similar documents for a linked document, even 27.3% are reached.

Cross-language Experiments – Model Variations

In Figure 6.7 and in Figure 6.8, we compare the full model with some model variations using the FA-4lang corpus. In all plots, the performance of the full model is depicted in gray to allow for an easy comparison. On the left side and on the right side of the plots, the percentage of document pairs being detected as similar within rank 1, rank 3, rank 10, and rank 50 are shown for the full model and for the model under analysis, respectively.

The performance of the basic model without the three features M, W, and N is shown in Figure 6.7(a). Due to the lack of the mapping feature, the basic model can only detect a similarity between two documents if both contain at least one identical event. Thus, the number of potentially similar documents for a given document decreases significantly. Considering the 23,710 documents containing at least one event, there are 281,070,195 document pairs (one-directional). Using the mapping feature, 15,001,220 similarity relations can be detected while only 680,816 similarity relations can be determined using model



(a) FM-MWN (basic model, BM).

(b) FM-WN (BM+M).

(c) FM-MN (BM+W).

(d) FM-MW (BM+N).

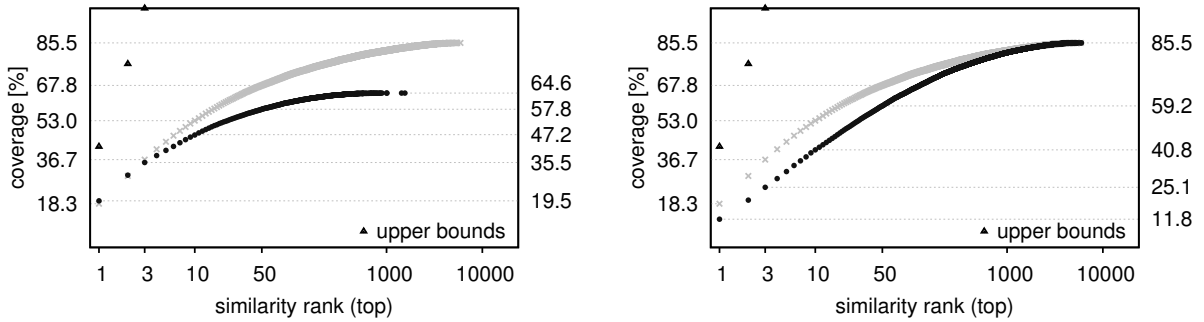
Figure 6.7: Comparing the evaluation results of the full model and three model variations, namely the basic model (a) as well as the basic model with the mapping feature (b), with the weighting feature (c), and with the normalization feature (d).

variations without the mapping feature. On average, 633 versus 29 similarity relations can be determined for each document. Even for the language-linked document pairs, instead of 85.5% with the full model, only 64.6% document pairs can be determined as similar with the basic model.

By adding the mapping feature to the basic model (BM+M), 85.5% of the document pairs can be determined as similar as shown in Figure 6.7(b). However, while a similarity can be determined for 85.5% of the document pairs, the BM+M model performs much worse than the full model with respect to the top 1000 similarity ranks.

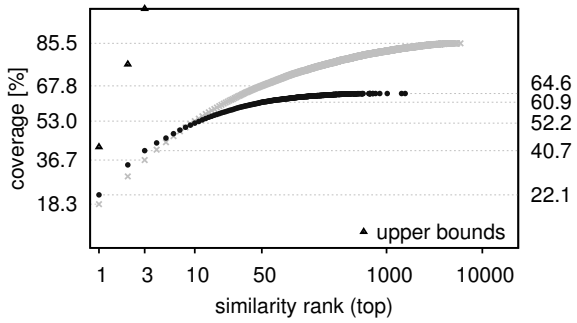
In contrast, when adding the weighting feature to the basic model instead of the mapping feature, the numbers for the top similarity ranks are even better than the numbers of the full model as plotted in Figure 6.7(c). Recall that the weighting factor is responsible to take into account the granularities of the events and makes fine-grained events, e.g., day-city events become more important than more coarse-grained events. Thus, Figure 6.7(b) and Figure 6.7(c) demonstrate the positive influence of the mapping feature (more similarity relations) and the weighting feature (better top ranks) on the model.

In Figure 6.7(d), the evaluation results for the basic model with the normalization feature added (BM+N) are presented. Although BM+N performs worse than the basic model on each rank, and although it does not help to increase the number of possibly similar document pairs as the mapping feature, we will show the positive influence of the normalization feature in combination with the other features.

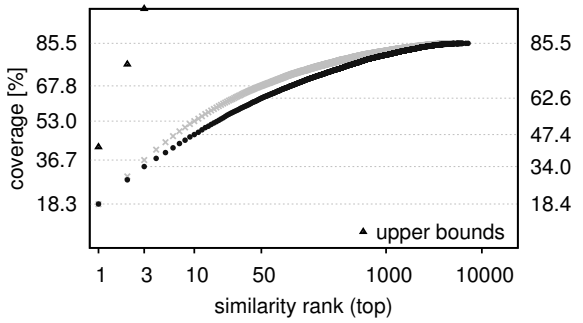


(a) FM-MWN (basic model, BM).

(b) FM-W (BM+M+N).



(c) FM-M (BM+W+N).



(d) FM-N (BM+M+W).

Figure 6.8: Comparing the evaluation results of the full model and three further model variations, namely the basic model with the mapping and the normalization features (b), with the weighting and the normalization features (c), and with the mapping and weighting features (d).

In Figure 6.8, we compare further model variations with the full model and with the basic model. We show thus again the performance of the basic model in Figure 6.8(a) for better comparability. The performance of the basic model with the mapping and the normalization features is presented in Figure 6.8(b). Except for rank 1, the model achieves better results than the basic model with the mapping feature but without the normalization feature (cf. Figure 6.7(b)). Similarly, the basic model with the weighting and the normalization features shown in Figure 6.8(c) outperforms the basic model with the weighting feature but without the normalization feature (cf. Figure 6.7(c)). This demonstrates the importance of the normalization feature as does the performance of the final model variation – the basic model with mapping and weighting features – depicted in Figure 6.8(d). Here, we can directly see that adding the normalization feature, which results in the full model, further boosts the performance of the model.

In summary, the comparison between several model variations demonstrates the effectiveness of all three features. Next, we study the performance of the full model with respect to the documents' categories.

Cross-language Experiments – Analysis of Document Categories

As mentioned above, we do not aim at developing a universal similarity model that is equally suitable for all kinds of documents. Since we fully rely on spatio-temporal events for calculating document similarity, we aim at finding valuable similarity relations for documents in which spatio-temporal events play a

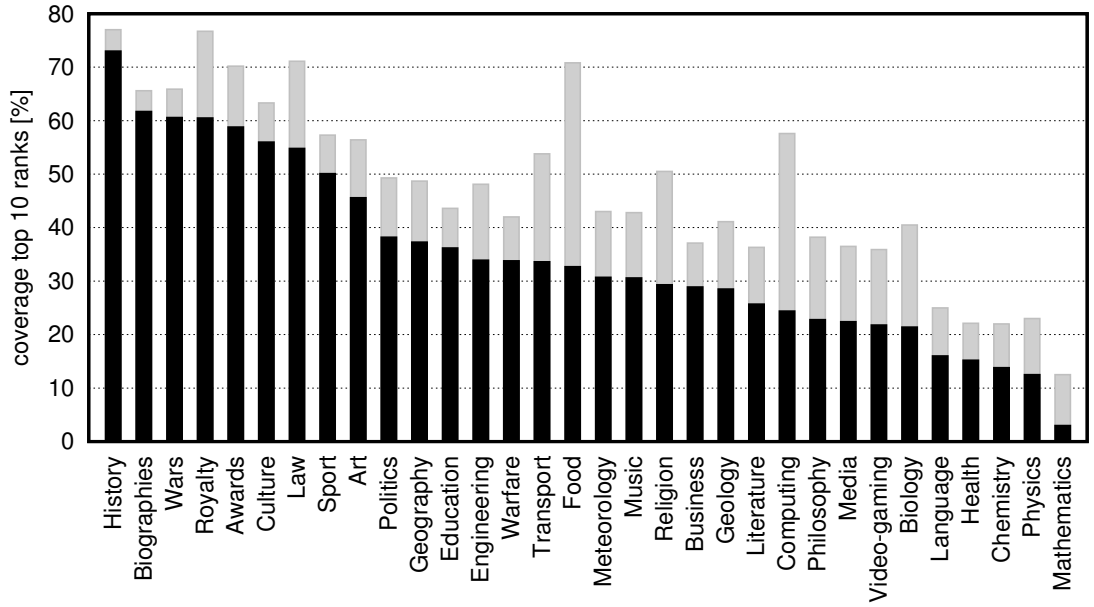


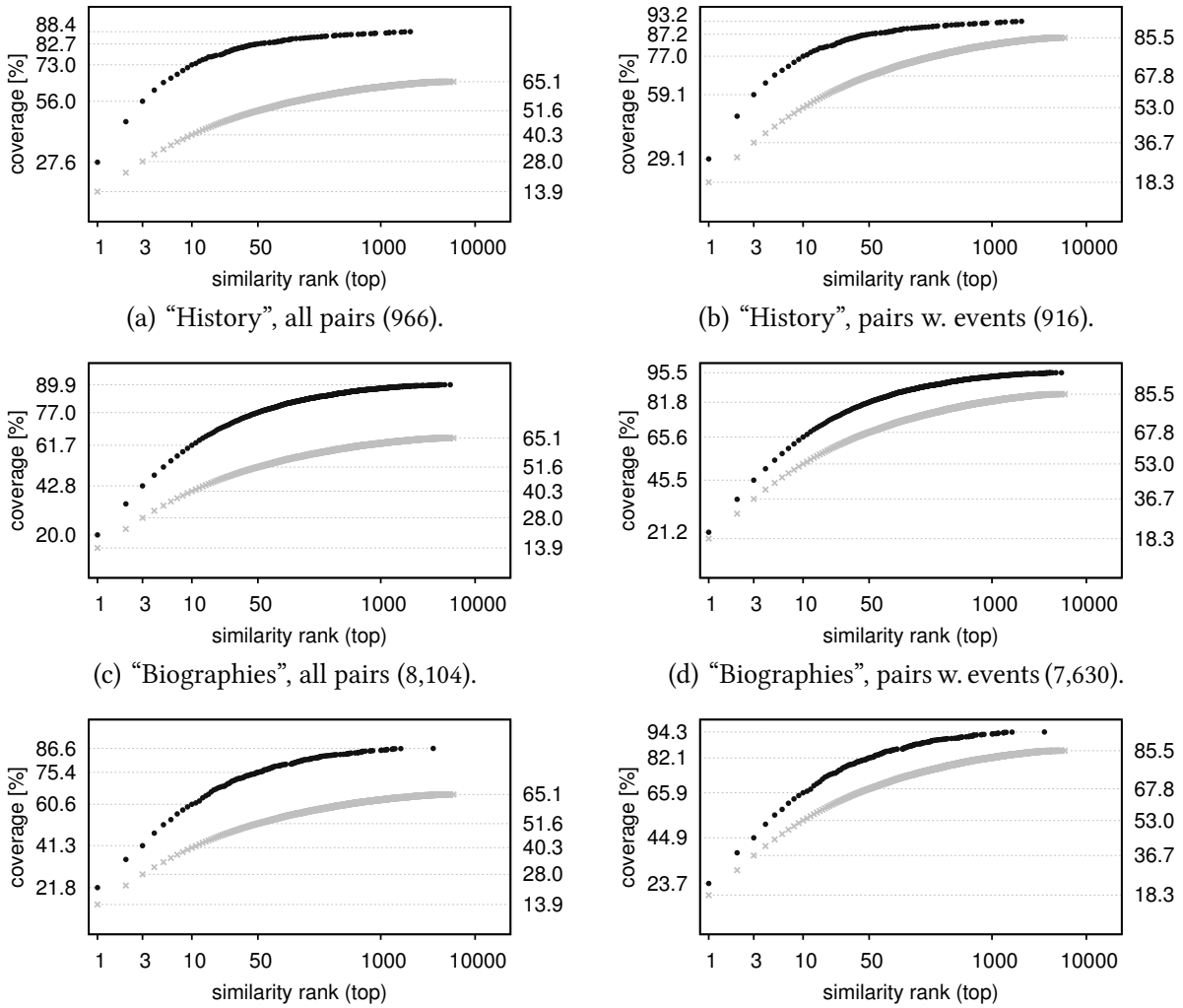
Figure 6.9: Comparing the cross-language evaluation results of all categories of the FA-4lang corpus. The gray bars show the rank 10 coverage of language-linked documents with considering only the document pairs for which both documents contain at least one event. The black bars show the rank 10 coverage of language-linked documents with respect to all documents of the categories.

central role, e.g., in documents about history or persons. Thus, we separately determine for each document category, how often language-linked documents are amongst the most similar documents for each other.

In Figure 6.9, we show the percentage of how often language-linked documents are within the top 10 most similar documents for all documents of the same category. Since there are different upper bounds for rank 1 and rank 2, and since we assume that some other documents can be even more similar than a language-linked document, we chose rank 10 for ordering the categories and not rank 1 or another lower rank. Furthermore, some categories contain many document pairs where at least one document does not contain any event, while other categories contain almost only document pairs where both documents contain events (cf. Table 6.4, page 237). Thus, we show rank 10 coverage with respect to the total number of document pairs (bidirectional) and with respect to the number of document pairs with at least one event per document in Figure 6.9 using black and gray bars, respectively.

The ordering of the categories is quite similar independent of which document set is considered although there are some exceptions, e.g., “food”, and “computing”. In these categories, the count of document pairs for which both documents contain at least one event is very low, however, if both contain events, these are characteristic.

In general, the event-centric document similarity model detects the similarity between language-linked documents particularly well for categories, such as “history”, “biographies”, “wars”, “royalty”, “awards”, “culture”, and “law”. In contrast, for documents about “mathematics”, “physics”, “chemistry”, “health”, “language”, and “biology”, only few language-linked documents can be detected as being similar. However, these results are intuitive since spatio-temporal events are characteristic for documents of the first group of categories while they are rather less characteristic for documents of the second group of categories.



(a) "History", all pairs (966).

(b) "History", pairs w. events (916).

(c) "Biographies", all pairs (8,104).

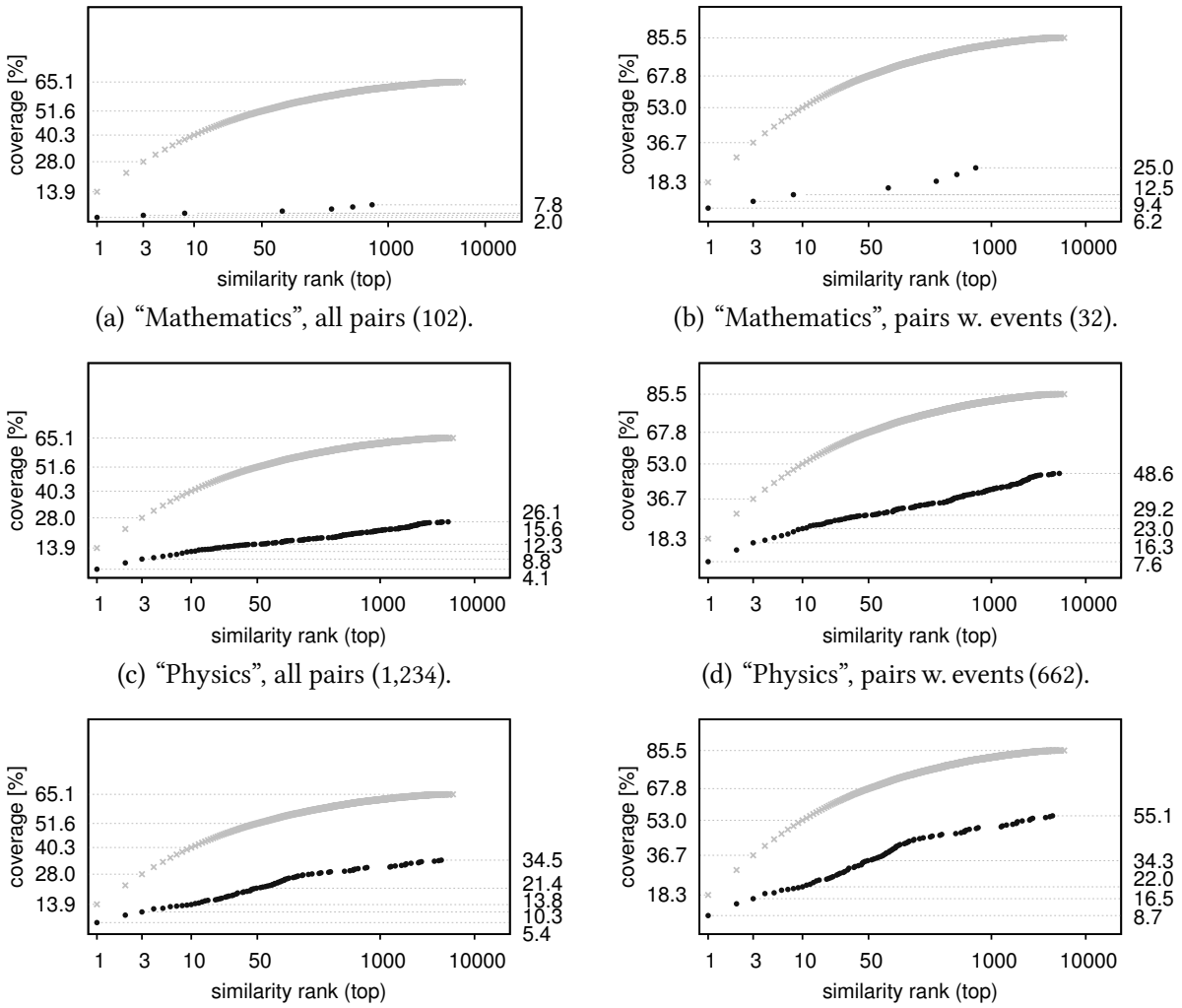
(d) "Biographies", pairs w. events (7,630).

(e) "Wars", all pairs (1,250).

(f) "Wars", pairs w. events (1,148).

Figure 6.10: Results of the cross-language evaluation for the top three categories "history" in (a) and (b), "biographies" in (c) and (d), and "wars" in (e) and (f), and the results on all categories for comparison (gray). Data sets for the left plots are all document pairs of the respective categories and for the right plots all respective document pairs containing events.

To deeper analyze the distribution of similarity ranks for the best performing and worst performing categories, we present the results for the three top performing categories and the three worst performing categories in more detail in Figure 6.10 and Figure 6.11, respectively. Similar to the results of the full model and model variations, we show in Figure 6.10(b), Figure 6.10(d), and Figure 6.10(f) the ranks of language-linked documents for each other for the categories "history", "biographies", and "wars" using as set of documents all document pairs for which both documents contain at least one event. In addition, we show in Figure 6.10(a), Figure 6.10(c), and Figure 6.10(e) the results using all document pairs for the three categories. For comparison, the results of all categories are also plotted in each figure (gray).



(a) "Mathematics", all pairs (102).

(b) "Mathematics", pairs w. events (32).

(c) "Physics", all pairs (1,234).

(d) "Physics", pairs w. events (662).

(e) "Chemistry", all pairs (406).

(f) "Chemistry", pairs w. events (254).

Figure 6.11: Results of the cross-language evaluation for the worst three categories "mathematics" in (a) and (b), "physics" in (c) and (d), and "chemistry" in (e) and (f), and the results on all categories for comparison (gray). Data sets for the left plots are all document pairs of the respective categories and for the right plots all respective document pairs containing events.

In the data sets with only those document pairs with events, about 95% of the possible similarity relations are detected in the three categories. Furthermore, the numbers for rank 10 are 73%, 61.7%, and 60.6% on all document pairs and 77%, 65.6%, and 65.9% on the document pairs with events. Thus, the results demonstrate the effectiveness of the event-centric similarity model for documents of those categories, for which spatio-temporal events play an important role.

In contrast, the results on the worst three categories (Figure 6.11) show that event-centric document similarity can hardly be determined between language-linked documents of categories for which spatio-temporal events are not characteristic. Many documents do not contain any spatio-temporal events, but even those pairs for which both documents contain events can only rarely be determined as similar.

Cross-language Experiments – Analysis of Language Pairs

As a final experiment using the FA-4lang corpus, we analyze the performance of the similarity model for each language pair separately. In Figure 6.12, the results are depicted. As in the previous plots, the performance on all document pairs is shown (gray) together with the results on the document pairs under analysis (black). This allows for an easy comparison.

Obviously, the results are quite different depending on the languages that are involved. However, the differing results are not due to the languages themselves but due to the different amounts of events that are extracted in the respective languages. As was shown in Table 6.4 (page 237), the average amount of events is much higher for English than for the other languages while it is lowest for German. In addition, the events are less well distributed among the categories for German than for Spanish and French due to the outlier category "sport, recreation". Thus, there are many German documents with only very few events.

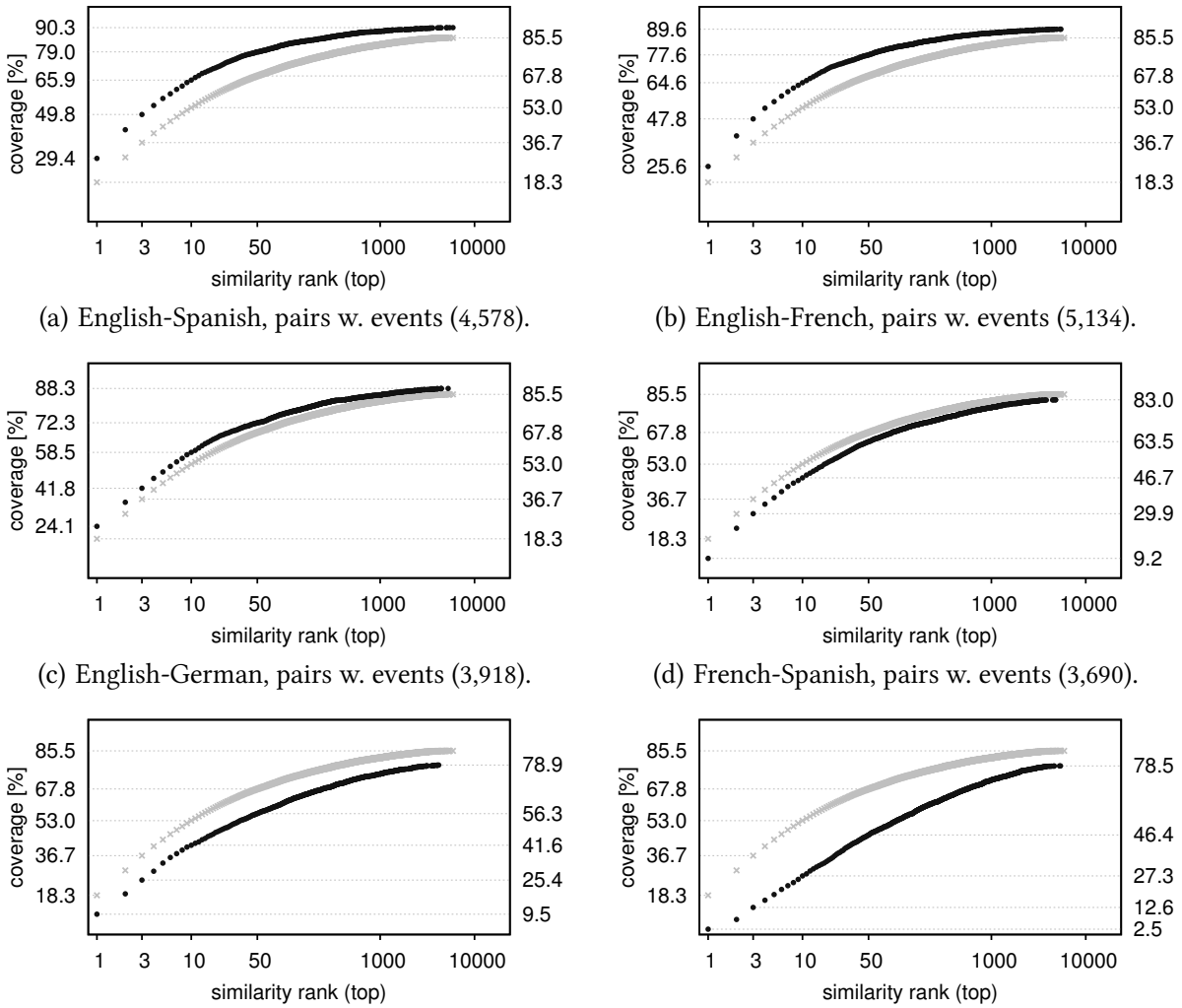
While the three language pairs with English included perform better than the other three language pairs, the best results are achieved for English-Spanish followed by English-French as shown in Figure 6.12(a) and Figure 6.12(b), respectively. For these two language pairs, for almost 50% of the documents their cross-language linked document in the respective language is ranked among the three most similar documents. For English-German, this statement still holds for almost 42% of the documents as depicted in Figure 6.12(c) while on the full data set the rank 3 result is 36.7%.

The results for French-Spanish depicted in Figure 6.12(d) are still good and only slightly worse than on the full data set. In contrast the results shown in Figure 6.12(e) and in particular those shown in Figure 6.12(f), i.e., for German-French and German-Spanish, are much worse. To achieve better results for these language pairs, the document similarity model would have to return higher similarity scores if both documents contain only few events. However, one of the requirements for the model defined in Section 6.5.4 was that it should not be penalized if only one document contained non- (or hardly) similar events additionally so that the event quantity normalization feature depends only on the number of events of the document with less events.

As it was shown above, this normalization strategy performs well in general although it becomes rather unlikely that two documents with only few events are determined as very similar if many documents with many more events also exist in the document collection. Note, however, that if the corpus contained only the German and Spanish (or the German and French) documents, the rank distribution for language-linked pairs would be much better because many long documents were excluded and removed from the set of potentially similar documents.

Comparison to Term-based Similarity Measure

The goal of comparing document similarity relations determined with the event-centric model to standard term-based similarity relations is not to demonstrate that the event-centric model outperforms term-based models. As described above, determining similarity is a subjective task, and documents can be similar to each other with respect to several aspect. One of these aspects is event-based similarity and by the comparison to a term-based similarity model, we want to demonstrate that an event-centric model and a term-based model determine different documents as being similar. As representative for term-based similarity measures, we select tf-idf combined with cosine similarity denoted $d\text{-}sim_t$ in the following.



(a) English-Spanish, pairs w. events (4,578).

(b) English-French, pairs w. events (5,134).

(c) English-German, pairs w. events (3,918).

(d) French-Spanish, pairs w. events (3,690).

(e) German-French, pairs w. events (3,326).

(f) German-Spanish, pairs w. events (3,064).

Figure 6.12: The results of the cross-language evaluation for each language pair separately. In each plot, the results on the full FA-4lang corpus are also depicted (gray).

To evaluate the differences between event-centric document similarity ($d\text{-}sim_e$) and $d\text{-}sim_t$, we analyze pairs of documents (d_i, d_j) according to their ranks for both scores. This results in four categories:

c1. (d_i,d_j) are similar for $d\text{-}sim_t$, but not for $d\text{-}sim_e$

c2. (d_i,d_j) are similar for $d\text{-}sim_e$, but not for $d\text{-}sim_t$

c3. (d_i,d_j) are similar for both scores

c4. (d_i,d_j) are not similar for either scores

This evaluation is again performed with the FA-4lang corpus, however, for each language separately. This ensures that the differences do not occur because the term-based similarity model prefers

to determine documents of the same language as similar while the event-centric similarity model is language-independent. We use the top-n ranked documents for $d\text{-}sim_e$, with $n \in \{1, 3, 5, 10\}$, i.e., $\text{rank}_e(d_i, d_j) \leq n$, and calculate the ratio of documents that are similar using $d\text{-}sim_t$ at each rank_t.

In Figure 6.13, these ratios are presented. In the left plots, all rank_t are depicted for the four rank_e values. In the right plots, the top 10 rank_t are shown in detail. Note that the larger rank_e, the more often a document pair is also within the top rank_t (represented as bars in the right plots of Figure 6.13) but the ratio is smaller (depicted as points). For instance, when considering all document pairs with $\text{rank}_e = 1$, we analyze x document pairs, but when considering $\text{rank}_e \leq 5$, $5 \times x$ document pairs have to be considered.

As depicted in the four right plots, for English, German, French, and Spanish, only 9%, 7%, 7%, and 6% of document pairs with rank_e = 1 are also determined as most similar with the term-based similarity model. Although these ratios increase to 22%, 18%, 17%, and 20% for English, German, French, and Spanish, respectively, when considering all document pairs ranked within the top ten according to both, $d\text{-}sim_e$ and $d\text{-}sim_t$, this experiment clearly demonstrates that using the event-centric document similarity model leads to the discovery of new similarity relationships, which are hidden in the event information of the documents. These cannot be discovered using standard similarity measures.

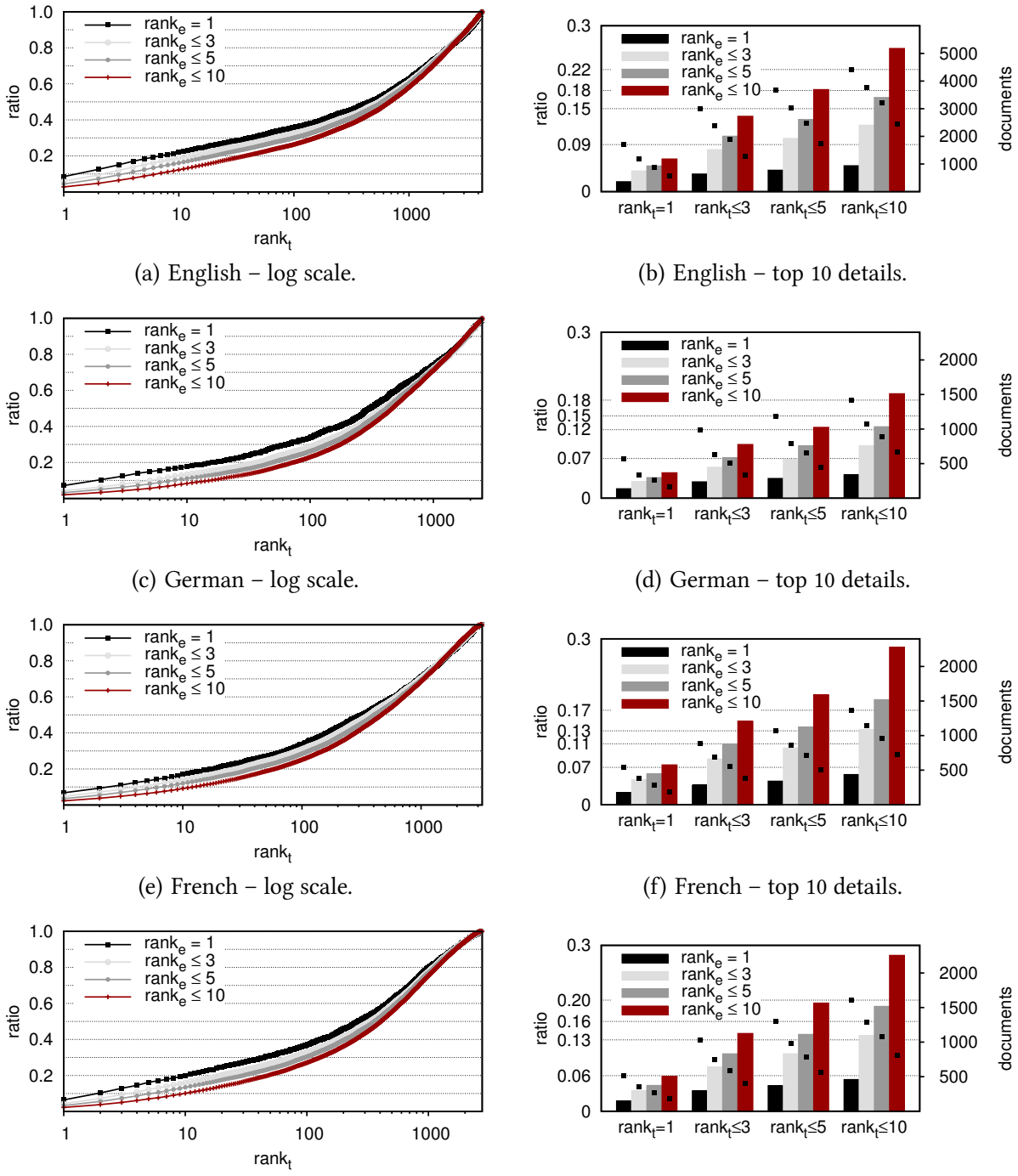
To demonstrate that similarity detection with both measures is valuable, we give examples of pairs of documents for the categories (c1) to (c3). As reference document d_1, for which similar documents are analyzed, we use the featured article "7 World Trade Center". It covers several sub-topics: the construction of the original building in 1987 in the center of New York, a description of the new building, and as major topic the collapse of the original building in the context of the September 11, 2001 terrorist attacks.

Some of the documents d_j of category (c1) with $\text{rank}_t(d_1, d_j) < 10$ and $\text{rank}_e(d_1, d_j) \gg \text{rank}_t(d_1, d_j)$, with "$\gg$" indicating "much larger than", are "Chicago Board of Trade Building" (rank_t=2, rank_e=415), "Manadnock Building" (rank_t=3, rank_e= –), "Trump International Hotel and Tower (Chicago)" (rank_t=4, rank_e=151), and "Scottish Parliament Building" (rank_t=5, rank_e=1672), i.e., articles about buildings, their construction and design. Although in some of these documents – mainly in those about buildings in the US – the 9/11 attacks are also mentioned, these play a minor role since these buildings were not affected.

In contrast, two of the documents d_k of category (c2) with $\text{rank}_e(d_1, d_k) < 10$ and $\text{rank}_t(d_1, d_k) \gg rank_e(d_1, d_k)$ are "Jihad (song)" (rank_e=1, rank_t=879) and "Wail al-Shehri" (rank_e=3, rank_t=1306). In both documents, the 9/11 terror attacks on the World Trade Center in New York play a major role – "[t]he song portrays the imagined viewpoint of a terrorist who has participated in the September 11, 2001 attacks",[6] and Wail al-Shehri was determined as one of the participating terrorist.

In addition, there are some documents that are less obviously related to d_1, e.g., "God Hates Us All" (rank_e=4, rank_t=2182), "Maggie Gyllenhaal" (rank_e=6, rank_t=1138) and "Bruno Maddox" (rank_e=7, rank_t=1845). While "God Hates Us All" is a studio album by Slayer, it was released on September 11, 2001 and is thus brought in connection to the terror attacks. "Maggie Gyllenhaal" is an American actress born in New York City, who played in Oliver Stone's movie "World Trade Center" – which is based on the 9/11 attacks in New York. Similarly, "Bruno Maddox" is a novelist who also published numerous articles in popular magazines – among them a famous article about "the callousness of the terrorists who flew into the World Trade Center".[7]

[6] `http://en.wikipedia.org/wiki/Jihad_(song)` [last accessed October 7, 2014].
[7] `http://en.wikipedia.org/wiki/Bruno_Maddox` [last accessed October 7, 2014].



(a) English – log scale.

(b) English – top 10 details.

(c) German – log scale.

(d) German – top 10 details.

(e) French – log scale.

(f) French – top 10 details.

(g) Spanish – log scale.

(h) Spanish – top 10 details.

Figure 6.13: Comparing event-centric and term-based similarity ranks on the FA-4lang corpus, for each language separately. Figures (a), (c), (e), and (g) show the ratio rank_e vs. rank_t for all document pairs and the $\text{rank}_e \in \{1, 3, 5, 10\}$ while details for the top 10 rank_t are depicted in (b), (d), (f), and (h); bars for total number of document pairs; points for ratio.

article	$rank_e$	$rank_t$	explanation
Jihad (song)	1	879	Song about 9/11 attacks in New York. → event-centric similar documents.
American Airlines Flight 11	2	9	Flight that crushed into one of the World Trade Center (WTC) towers. → event-centric similar documents.
Wail al-Shehri	3	1306	Terrorist who flew into one of the WTC towers. → event-centric similar documents.
God Hates Us All	4	2182	Studio album released on Sept. 11, 2001 and brought in connection with the terror attacks. → event-centric similar documents.
Arbiter (Halo)	5	3661	Computer game character voiced by a New York City actor. The article contains only two spatio-temporal events, both with geographic component "New York City" and temporal components "2004" and "2007". This makes both events very similar to many events in d_1. However, although the 9/11 attacks are mentioned in the article, they are not extracted as spatio-temporal event. Thus, there is no obviously justifiable event-centric document similarity. → not event-centric similar documents.
Maggie Gyllenhaal	6	1138	Actress playing in the "World Trade Center" movie about the terror attacks. → event-centric similar documents.
Bruno Maddox	7	1845	Writer who wrote a famous article about the callousness of the terrorists attacks. → event-centric similar documents.

Table 6.6: The most similar documents to the English featured article "7 World Trade Center" with respect to $d\text{-}sim_e$ considering only the English subset of the FA-4lang corpus.

Finally, a document d_l of category (c3) is the article "American Airlines Flight 11" ($rank_e$=2, $rank_t$=9) about the plane that crushed into the World Trade Center – obviously similar to the document about "7 World Trade Center" both, topically similar as well as with an event-centric aspect.

For a more complete overview, Table 6.6 shows the most similar documents to the article "7 World Trade Center" according to sim_e, with brief explanations if and why the documents are similar in an event-centric way. For these similarity calculations, only the English documents of the FA-4lang corpus are considered. In addition, we show in Table 6.7 the most similar documents to the same article using the full FA-4lang corpus to demonstrate that the similarity model is language-independent and that event-centric similarity is detected across documents of different languages.

However, to demonstrate that these exemplarily-selected documents are not just exceptions, we will present in the following the results of our manual evaluation.

$rank_e$	article	language	manual judgment
1	Jihad (song)	French	→ event-centric similar documents
2	7 World Trade Center	German	→ event-centric similar documents
3	7 World Trade Center	French	→ event-centric similar documents
4	Jihad (song)	Spanish	→ event-centric similar documents
5	Wail al-Shehri	Spanish	→ event-centric similar documents
6	Jihad (song)	English	→ event-centric similar documents
7	American Airlines Flight 11	German	→ event-centric similar documents
8	American Airlines Flight 11	English	→ event-centric similar documents
9	7 World Trade Center	Spanish	→ event-centric similar documents
10	God Hates Us All	German	→ event-centric similar documents

Table 6.7: The most similar documents to the English featured article "7 World Trade Center" with respect to $d\text{-}sim_e$ considering the full FA-4lang corpus to also demonstrate valuable similarity relations between documents of different languages.

Manual Evaluation

The objective of the manual evaluation is to validate the precision of the event-centric similarity model. For this, we use the FA-4lang corpus and randomly select 40 articles from the categories history, wars, and biographies as source documents, 20 for English and 20 for German. In the detailed category-based analysis of the evaluation results described above, we showed that these categories are especially suitable for determining event-centric document similarity since in such documents events play a major role and occur frequently. Since we do not claim that event-centric similarity is useful for all types of documents, a selection of documents from those categories that usually contain documents with many spatio-temporal events is useful.

For each source document, we select the ten most similar documents[8] and evaluate if they contain at least one exactly same event that is not too coarse grained (e.g., World War II in general with references to a country and a year, e.g., Germany and 1945, would not be sufficient while "Japan's surrender in September 1945" would be sufficient). Since similarity is quite subjective in general, we set this rather strict criterion to ensure objectivity in the manual evaluation process. Nevertheless, even documents that do not contain any identical events could sometimes be considered as similar in an event-centric way.

For each document pair, that is evaluated as being similar in an event-centric way, we further determine (i) if they belong to the same category, (ii) if they have a similar main topic, (iii) if they are written in the same language, and (iv) if the two documents build a language pair. While all these further features contain rather general information, they will help to draw conclusions when analyzing the evaluation results. In addition, if two documents are not considered as similar, we also collect the main reasons and present at the end of this section typical reasons for incorrectly identified similarity relations.

In Table 6.8, we present the evaluation numbers for the 20 English and the 20 German source documents separately. In particular, we show the precision at k with $k \in \{1, 2, 3, 5, 10\}$ indicating how many of

[8] In the initial evaluation described in (Strötgen et al., 2011), we also used 40 documents but checked only the top 5 ranked documents for each source document. In addition, the initial evaluation was performed on the old Wiki Featured Articles corpus containing only documents of two languages. In contrast, this new manual evaluation is now performed on the new FA-4lang corpus described in Section 6.5.7.

(a) Manual evaluation using 20 English source documents.

	same event(s) (d-sim$_e$ is true)	same category*	same main topic*	same language*	language pair*	lang. pair coverage
p@1	**0.9**	0.94	0.89	0.0	0.78	14 / 19
p@2	**0.85**	0.91	0.85	0.09	0.71	24 / 37
p@3	**0.77**	0.91	0.89	0.11	0.70	32 / 53
p@5	**0.71**	0.77	0.76	0.15	0.49	35 / 53
p@10	**0.64**	0.69	0.61	0.22	0.32	41 / 53

(b) Manual evaluation using 20 German source documents.

	same event(s) (d-sim$_e$ is true)	same category*	same main topic*	same language*	language pair*	lang. pair coverage
p@1	**0.95**	0.95	0.95	0.11	0.79	15 / 20
p@2	**0.93**	0.95	0.89	0.11	0.65	24 / 39
p@3	**0.85**	0.96	0.88	0.10	0.59	30 / 55
p@5	**0.74**	0.91	0.84	0.11	0.49	36 / 55
p@10	**0.63**	0.84	0.74	0.11	0.34	43 / 55

Table 6.8: Manual evaluation using 20 English (a) and 20 German (b) source documents. Precision at 1, 2, 3, 5, and 10 values measure how many of the retrieved documents are similar in an event-centric way. The further features are determined for correctly as similar identified documents (*). The language pair coverage represents how many of the possibly retrieved language-linked documents are retrieved.

the k most similar ranked documents for a given source document are not only detected by the similarity model but also manually evaluated as being similar in an event-centric way (cf. Equation 2.9, page 31). As shown in Table 6.7(a), the values for precision at 1, 2, and 3 are 90%, 85%, and 77%, and decrease to 71% and 64% for precision at 5 and 10, respectively. The results for German are slightly better, with precision values of 95%, 93%, 85%, 74%, and 63% at the five recall levels. In general, the results for both languages are quite sophisticated.

One difference between the English and the German experiments is that there is for each German article at least one language-linked article in the document collection – the respective English document – but for one of the randomly selected English articles there was no language-linked article. Furthermore, for 19 of the German documents, there are at least two language-linked articles in the document collection and for 16 documents there are even three, while for 18 and 16 English documents two and three language-linked documents exist. Among the ten most similar documents for the English and German source documents, there are 41 of the 53 and 43 of the 55 language-linked documents, respectively.

Analyzing the "same category" and "same topic" features indicates that many of the correctly as similar detected articles belong to the same category as the source document and describe a similar main topic. This is particularly true for the top 3 ranked documents. Note, however, that for each language-linked document, the category and the main topic are obviously identical. In contrast, they are written in one of the other languages of the corpus, which is a main reason why the "same language" feature is quite low. In addition, this low value shows that the similarity detection process is independent of the documents' languages.

Manual Evaluation – Error Analysis

Finally, we briefly discuss the main reasons why documents have been incorrectly determined as similar to a source document. For some articles, there are probably not many other articles in the document collection that can be considered as similar in an event-centric way. Given the heterogeneity of the corpus, this is not surprising – in particular if a source document deals with a rather special and specific topic.

However, there are also two main reasons for errors that are due to the similarity model or the settings that were chosen for the experiments. The first issue is that given a document containing only very few spatio-temporal events, it is quite likely that documents containing extraordinary many events are determined as similar although only a rather small ratio of the occurring events are only slightly similar to the source documents' few events. This is due to the selected event quantity normalization feature which only considers the quantity of events of the document with the lower number of events. Although this normalization procedure was used intentionally due to one of the model requirements,[9] it results in relatively high similarity scores for document pairs for which one document contains only very few and the other very many events. For instance, for those of the German source documents with only few events, some documents with huge amounts of events coarsely fitting to the time and place of the source documents' event(s) are among the top ranked documents.

Similarly, for those of the English source documents with very many events, there have been some single-event documents in the similarity results list since these benefit of the normalization process. Thus, in particular when being faced with a very heterogeneous corpus such as the FA-4lang corpus with very differing amounts of events per document, an adaptation of the normalization might increase the precision of the search results. Note, however, that the overall results are very good which is particularly demonstrated by the high amount of language-linked documents among the top similarity ranks.

A second error source that we detected when manually evaluating the document pairs occurs when the source documents contain only rather coarse events, e.g., with year and country granularities. While coarse events are not problematic when dealing with documents about long-time ago history because such documents are typical quite similar with respect to mentioned events if they cover the same time period and geographic area, they become an error source when dealing with more recent topics. Obviously, documents about relatively recent happenings on a country and year level can be quite different from each other with respect to mentioned events. Thus, for these source documents several high ranked documents also contain spatio-temporal events with similar temporal and geographic information but often also on a coarse level so that such documents cannot be considered as similar in an event-centric way – in particular in our evaluation for which we require at least a non-coarse event to occur in both documents. To avoid such errors, one could specify that only fine-grained events are considered during the event similarity calculation process, although many meaningful and correctly detected similarity relations between documents would probably disappear.

Finally, another reason for incorrectly identified similarity relations are errors in the event extraction process and in particular in the geo-tagging process. Although these errors did not occur frequently in our new experiments, geo-tagging errors were a major error class in our initial study (Strötgen et al., 2011) – in particular due to person names being tagged as locations.

[9] Requirement A3, as mentioned in Section 6.5.4: "If only one document contains additional events, this should not be penalized as much as if both documents contain additional non-matching events."

Summary

In summary, we have been able to demonstrate that documents that can be assumed to be similar in an event-centric way – i.e., cross-language linked documents – have often been detected as being similar using our event-centric similarity model and the FA-4lang corpus as evaluation data. In addition, this was also shown on the larger Wiki-XML corpus. Furthermore, we demonstrated that the model's single features are useful and lead to improved evaluation results by performing a detailed feature analysis using the FA-4lang corpus. Finally, by analyzing the evaluation results based on the documents' category information, we determined several categories for which the event-centric document similarity model works particularly well, e.g., biographies, history, wars, but also culture and sports.

The comparison with a term-based similarity model showed that other types of similarity are identified by our model compared to those detected by a standard term-based model. Finally, the manual evaluation demonstrated that not only detected similarity relations between cross-language linked documents are meaningful, but also many of the similarity relations to other documents that the model ranked as being similar. Thus, the event-centric document similarity model reveals meaningful similarity relations and can be applied whenever event-centric similarity may be of interest to explore document collections.

6.6 Further Types of Event-centric Similarity

In this section, we briefly outline two further approaches that are based on event similarity.

6.6.1 Event-centric Person Similarity

While it is often valuable to exploit event-centric document similarity to explore document collections, we already explained in Section 6.3 that instead of focusing on documents, it is also possible to directly focus on the events. A similar idea is that events mentioned in documents cannot only be associated with the documents but also with other named entities occurring in the context of the events. In particular, there are many types of documents in which many events can be associated with persons so that a personalized event profile can be created for each person.

Thus, we can also determine event-centric person similarity by using all the events that can be enriched with person information and by calculating the similarity based on personalized event profiles instead of event document profiles. Although there are of course many aspects with respect to which two persons can be considered as similar (e.g., size, weight, gender, etc.), the motivation for event-centric person similarity is that a person can be well characterized by the events he or she was or will be involved in. Thus, similarity between persons can be determined based on those events as well.

With "EvenPers" (Kapp et al., 2013), we presented a prototype to serve a very frequent search activity on the Web, namely searching for people. Searching for a person with the EvenPers system results in a set of events that are associated with that person. In addition to showing these events on a map and presenting them as trajectories, it is further possible to select an event to retrieve other persons with which the same event is also associated. Furthermore, a list of most similar persons to the initial person is also presented based on pre-calculated event-centric person similarity scores.

Obviously, instead of searching for persons solely by their names, temporal and geographic constraints could also be specified, and these can be validated based on the personalized event profiles. All the

functionality presented in Chapter 5 – such as query interfaces and indexing of events – could be directly applied to search for persons instead of documents.

6.6.2 Adaptation of the Similarity Model to the Biomedical Domain

As a final variation of the event-centric document similarity model, we briefly present its adaptation to the biomedical domain. In general, the biomedical domain is a very active research area and the number of publications increases rapidly. Often, events such as protein-protein interactions play an important role so that there is a lot of research on automatically extracting biomedical entities and events from documents. Thus, automatically extracted event information can be used to identify documents in the biomedical literature, which are similar to each other in an event-centric way.

Obviously, spatio-temporal events and biomedical events are quite different (cf. Section 4.2.3) so that some adaptations to the similarity model are required. The temporal and geographic components of the events have to be replaced by typical biomedical event components. Since these are however also organizable in a hierarchical structure, the calculation of event similarity can be performed in a quite similar way for biomedical events as for spatio-temporal events.

In (Keller et al., 2012), we showed that with some adaptations, the event-centric document similarity model can be used with biomedical events and to determine document similarity relationships in biomedical literature. For this approach, we relied on the GENIA definition of a biomedical event (Kim et al., 2006), so that an event can have up to two "themes", and/or up to two "causes", and one "event-type". While a "theme" component is a biological entity whose properties are changed by an event, and a "cause" component is a biological entity, which affects the way of occurrence of an event, the "type" component represents the biomedical relationship (e.g., binding or phosphorylation). Note that each event component can be normalized and can be associated with a concept in a hierarchy, namely within the GENIA term ontology and the GENIA event ontology (Kim et al., 2008). Thus, they share important key characteristics with temporal and geographic information, which are crucial for our event-centric document similarity model.[10]

6.7 Summary of the Chapter

In this chapter, we developed and explained several event-centric search and exploration approaches. The idea of event-centric search is that a document collection can be queried based on textual, temporal, and geographic constraints. However, in contrast to spatio-temporal search, the temporal and geographic constraints are directly used to search for spatio-temporal events satisfying these constraints and thus not validated separately. The further process, i.e., the retrieval process, can be addressed from two sides: (i) documents can be returned based on the relevance of the events they are containing, or (ii) events can be directly returned – similar as in entity-oriented search.

For the exploration of the search results – independent of whether documents or events are returned to a user – we presented several map-based approaches. In particular, the concepts of document trajectories and event sequences have been introduced, which are combined visualizations of the temporal and geographic information of multiple events. Event snippets have been developed to facilitate a smooth event-centric exploration of the search results.

[10] For further details to the adaptation of the model and evaluation results, we refer to (Keller et al., 2012).

A further major task addressed in this chapter is event-centric document similarity. For their detection, we first explained how the similarity between single spatio-temporal events can be calculated based on their normalized temporal and geographic information and the underlying temporal and geographic hierarchies. This procedure allows to detect not only similarity relations between documents containing identical events, but also if they contain events that are "only" similar. Furthermore, the model is term- and language-independent and can thus be used to detect similarity relations in multilingual corpora and across multiple languages. In an extensive evaluation, we showed that the model's features are effective to determine event-centric document similarity, in particular for those types of documents that have have been proven to be rich in event information, e.g., documents about biographies and history.

Finally, we outlined some further approaches relying on the event similarity model, e.g., to search for persons with a focus on events they participated in and to detect event-centric similarity relations between persons.

7 Conclusions and Future Work

Temporal and geographic information is important and ubiquitous in many types of documents, and often, it is used to describe events, e.g., in documents about history and biographies. This naturally results in the fact that temporal, geographic, and event-centric information needs are also frequent. However, such information needs have not been well served yet. Thus, we developed spatio-temporal and event-centric search and exploration frameworks. In this final chapter of the thesis, we first summarize the key aspects of our work and present our concluding remarks. Then, in Section 7.2, we discuss open issues and suggest directions for future research.

7.1 Summary and Conclusions

After motivating spatio-temporal and event-centric information retrieval, we detailed the main challenges for addressing these topics and explained our main contributions. In Chapter 2, we placed the thesis into its general research context and introduced the most important basic concepts, which laid the foundations for the remainder of the thesis, e.g., we explained the tasks of named entity recognition and named entity normalization, as well as the key characteristics of temporal and geographic information.

A prerequisite for spatio-temporal and event-centric information retrieval is the extraction and normalization of temporal and geographic expressions from documents. In our work, we relied on an existing geo-tagger, but addressed temporal tagging due to the lack of suitable, publicly available tools for this task. ***Chapter 3 covered this topic of temporal tagging*** and contained several important contributions. We started with a detailed survey of related work on annotation standards, research competitions, annotated corpora, as well as approaches to temporal tagging. Furthermore, we gave an overview of available temporal taggers and pointed out the most important open issues that are responsible for the fact that no multilingual, domain-sensitive temporal tagger existed which we could have used.

Thus, we addressed these issues and studied ***multilingual and cross-domain temporal tagging***. We analyzed domain-specific challenges, created manually annotated non-standard domain corpora, and developed domain-sensitive temporal tagging strategies – our first important contributions of Chapter 3. Then, related approaches to multilingual temporal tagging were surveyed, and our contributions to multilingual temporal tagging were presented, e.g., newly created annotated corpora and a description of the challenges of temporal tagging of different languages.

As a further major contribution, we developed and explained the ***design and implementation of our temporal tagger HeidelTime***. In extensive evaluations, we demonstrated HeidelTime's high extraction and normalization quality across all addressed languages and domains. By making HeidelTime publicly available as an easy-to-extend, multilingual, domain-sensitive temporal tagger, we made further important contributions. Besides state-of-the-art extraction and normalization quality, HeidelTime is particularly valuable because it is the only publicly available temporal tagger for several languages. It is thus used by several research groups worldwide.

Next, in ***Chapter 4***, we laid the theoretical ***foundations for event-centric information retrieval*** by introducing the concept of spatio-temporal events. We first surveyed event concepts of other research areas and analyzed their characteristics. Based on this analysis, we concisely defined our spatio-temporal events. In addition, we introduced several further concepts to store and "compute" with temporal, geographic, and event information. Finally, we discussed and compared heuristic and linguistically-motivated approaches to extract spatio-temporal events from text documents. Although, the rather simple cooccurrence approach achieved good evaluation results, we showed that the precision of the event extraction process can be improved by applying more sophisticated extraction methods.

By covering the concept of spatio-temporal events, Chapter 4 enabled us to distinguish our work from many related approaches surveyed in this thesis. In contrast to those approaches, we do not consider geographic and temporal information in isolation but explicitly combine temporal and geographic expressions into meaningful information nuggets. The further developed concepts were crucial for event-centric search and exploration tasks and also for performing spatio-temporal information retrieval that was tackled in Chapter 5.

In general, temporal and geographic information needs are not well served by standard search engines, and the reasons for this are twofold. First, it is often difficult to formulate temporal and geographic constraints in a meaningful way since a query is usually only a text string. Second, temporal and geographic expressions occurring in documents are regarded as standard terms so that their temporal and geographic meaning is lost. In ***Chapter 5***, we addressed these issues. After a detailed survey of related work on temporal, geographic, and spatio-temporal information retrieval, we introduced a ***multidimensional query model*** that allows to combine a text query with temporal and geographic constraints.

As a further main contribution, we introduced a retrieval model to rank documents based on how well they satisfy the parts of a multidimensional query. For this, the temporal and geographic expressions of all documents of a document collection have to be extracted, normalized, and stored in a preprocessing step. Using the concepts introduced in Chapter 4, query intervals and regions can be compared to the expressions occurring in the documents. Exploiting this information, we developed a ***relevance measure combining temporal, geographic, and textual relevance***. Finally, efficient indexing strategies and an evaluation of the ranking model were also presented in Chapter 5.

An important difference to other approaches to temporal, geographic, and spatio-temporal retrieval models is that ***our approach considers two types of proximity information***. First, the text proximity of terms satisfying the three query dimensions is determined, and small distances are rewarded. Second, if a document does not contain any temporal or geographic expressions satisfying the query intervals and regions, the temporal and geographic proximity of occurring expressions to the query intervals and regions are determined, and small distances are again rewarded. With the first proximity feature, we eliminated the assumption of previous approaches that different query dimensions are independent, and with the second proximity feature, the recall in spatio-temporal information retrieval can be increased.

In ***Chapter 6***, we developed and demonstrated several ***event-centric search and exploration frameworks***. In particular, we adapted our spatio-temporal retrieval model to directly query for events. Search results can thus be organized as ranked list of documents or as ranked list of events. Event snippets were introduced as a further general exploration feature. Additional main contributions are our frameworks for map-based exploration of documents and document collections, e.g., we introduced document trajectories

and event sequences which enable a user to explore temporal and geographic information of events in a combined way.

As final major contribution of the thesis, we introduced an ***event-centric document similarity model*** for discovering new types of document similarity. By exploiting the key characteristics of events and by applying the concepts introduced in Chapter 4 to "compute" with temporal, geographic, and event information, similarity scores are first determined between single events. Then, event similarity information for all events of different documents is usefully aggregated so that document similarity scores can be determined solely based on event information. Our detailed evaluation on multilingual corpora demonstrated the usefulness of our approach, in particular when considering documents in which events occur frequently. In addition, the evaluation highlighted the term- and language-independence of the similarity model in particular, and of our spatio-temporal event concept in general.

7.2 Future Work

In this thesis, we addressed several research questions in the research domains of information extraction and information retrieval. Although we demonstrated the high quality and usefulness of our approaches, there are some interesting areas in which our work could be extended as detailed in the following paragraphs.

Temporal Tagging and HeidelTime Extensions

While HeidelTime is a multilingual temporal tagger, there are of course many languages which are not yet supported. In this thesis, we detailed how ***HeidelTime resources for further languages*** can be created. Furthermore, researchers of other institutes developed language resources and also published papers about how to do so (see, e.g., Moriceau and Tannier, 2014). This demonstrates the feasibility of developing HeidelTime resources even if one is not involved in HeidelTime's development and not initially familiar with HeidelTime's implementation details. In contrast to developing a new temporal tagger for a not yet supported language, time and effort are quite manageable when choosing to extend HeidelTime. Thus, we are confident that further language resources for HeidelTime will be developed in the future.

There had been some approaches in the past to automatically extend a temporal tagger to further languages. However, such approaches did not result in as high quality temporal tagging performance as reached by taggers specifically developed for the respective languages. Nevertheless, we think that there is some potential to ***(semi-)automatically develop*** or at least to ***(semi-)automatically improve language resources*** for HeidelTime. For instance, a parallel corpus could be exploited by running HeidelTime on all language parts for which resources already exist. All normalized temporal information not extracted from all parallel documents can then be hints that there are either false negatives in some languages or false positives in other languages.

Finally, ***automatic domain and language detection*** functionality could be added to HeidelTime. For language detection, standard tools could be integrated, and for domain detection, the results of our cross-domain corpus analysis could be exploited. Since the document creation time plays an important role for processing documents of some domains, in particular news-style documents, it should be provided to HeidelTime. For this, automatic document creation time detection should also be integrated. Recently,

Tannier (2014) suggested an approach for this task for English and French which could be extended to other languages and combined with automatic domain detection methods.

Event Extraction

Our spatio-temporal event extraction methods have been developed and tested with multilinguality in mind. In addition, the simple cooccurrence approach already provided good evaluation results. However, one could try to further ***improve the precision of the event extraction process***. In general, the task of deciding whether a cooccurrence of a geographic and a temporal expression forms a spatio-temporal event is a typical classification problem. Thus, one could use our heuristic features and add further linguistic features to train a machine learning classifier. In particular if one does not aim for multilingual solutions, linguistic features such as part-of-speech and lexical information could be helpful. However, the amount of manually annotated cooccurrences should be increased and training data should be available for all languages that shall be addressed.

Spatio-temporal Information Retrieval

We introduced a spatio-temporal information retrieval model that takes into account textual, temporal, and geographic relevance as well as proximity information between terms satisfying the different query dimensions. Based on this model, a spatio-temporal search engine could be developed for corpora collections in which geographic and temporal information plays an important role. For instance, using a constantly updated dump of Wikipedia would allow to set up a ***spatio-temporal search engine for Wikipedia*** with sophisticated query functionality and a promising ranking strategy to determine documents that are relevant with respect to the textual, temporal, and geographic constraints. Other suitable document collections would be static collections in the area of digital libraries. Such systems could then also be exploited to perform ***more detailed evaluations of our ranking approach***. In addition, since performance optimization was not a major topic in this thesis, one could test ***further indexing methods*** and compare them with our approach.

Event-centric Search and Exploration Frameworks

Our concept of spatio-temporal events did so far only consider temporal and geographic information. This has the advantage that both components of an event can be normalized and organized hierarchically, and all events are term- and language-independent. However, for visualizing search results and to explore spatio-temporal events, context information about the events could be exploited. For this, we already introduced event snippets which presented spatio-temporal events together with the sentences from which they were extracted. A simple approach to collect ***more meaningful event context information*** would be to add to each event a word vector with all words cooccurring with the events in the same sentences.

In addition, the contexts of single spatio-temporal event instances could be aggregated for each event. Then, for each event in the form of a temporal and a geographic component, corpus-wide context information could be explored. Events could be compared to each other not only based on their temporal and geographic components but additionally with respect to their context vectors. Obviously, the vectors could also be weighted using, e.g., tf-idf values calculated with respect to the context vectors of all events extracted from a document collection. The vectors could then be compared with each other using standard measures such as the cosine similarity.

As an alternative to adding all content words to each event, only words of interest, e.g., words organized in a special taxonomy of interest, could be considered. By using only a controlled vocabulary, the vectors would again contain normalized information so that the context vectors would be again language-independent. Another idea would be to combine only cooccurring named entities such as persons to each event. If cross-document coreference resolution is applied additionally, event profiles for all persons occurring in a document collection can be created. While we presented an initial prototype for event-person correlations, ***more advanced exploration scenarios*** could be realized to determine event-centric person similarity and more complex correlations between persons and events.

Summary

Although we presented many significant contributions in this thesis, we are – not only based on our above presented suggestions – optimistic that we simultaneously open new directions and perspectives for future research. To give just one example: Since temporal tagging can now be performed on documents of different domains and many different languages, temporal information can now be exploited in all kinds of multilingual research settings. Besides rather obvious research areas such as machine translation and multilingual information retrieval, even further areas of research that we have not yet considered might profit from this. We are optimistic that we contributed to the ambitious goal of full natural language understanding – not only on English news-style documents but on a more general level.

Bibliography

Charu C. Aggarwal and ChengXiang Zhai, editors. *Mining Text Data*. Springer, New York, NY, USA, 2012.

David Ahn, Sisay Fissaha Adafre, and Maarten de Rijke. Towards Task-based Temporal Extraction and Recognition. In Graham Katz, James Pustejovsky, and Frank Schilder, editors, *Annotating, Extracting and Reasoning about Time and Events*, number 05151 in Dagstuhl Seminar Proceedings, 2005.

James Allan. Introduction to Topic Detection and Tracking. In James Allan, editor, *Topic Detection and Tracking: Event-based Information Organization*, chapter 1, pages 1–16. Kluwer Academic Publishers, Norwell, MA, USA, 2002a.

James Allan, editor. *Topic Detection and Tracking: Event-based Information Organization*. Kluwer Academic Publishers, Norwell, MA, USA, 2002b.

James Allan, Rahul Gupta, and Vikas Khandelwal. Temporal Summaries of News Topics. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '01)*, pages 10–18. ACM, 2001.

James F. Allen. Maintaining Knowledge about Temporal Intervals. *Communications of the ACM*, 26(11): 832–843, 1983.

Omar Alonso. *Temporal Information Retrieval*. PhD thesis, University of California Davis, 2008.

Omar Alonso, Michael Gertz, and Ricardo Baeza-Yates. On the Value of Temporal Information in Information Retrieval. *ACM SIGIR Forum*, 41(2):35–41, 2007.

Omar Alonso, Michael Gertz, and Ricardo Baeza-Yates. Clustering and Exploring Search Results using Timeline Constructions. In *Proceedings of the 18th ACM International Conference on Information and Knowledge Management (CIKM '09)*, pages 97–106. ACM, 2009.

Omar Alonso, Jannik Strötgen, Ricardo Baeza-Yates, and Michael Gertz. Temporal Information Retrieval: Challenges and Opportunities. In *Proceedings of the 1st International Temporal Web Analytics Workshop (TWAW '11)*, pages 1–8. CEUR-WS.org, 2011.

Einat Amitay, Nadav Har'El, Ron Sivan, and Aya Soffer. Web-a-Where: Geotagging Web Content. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '04)*, pages 273–280. ACM, 2004.

Sophia Ananiadou and John McNaught, editors. *Text Mining for Biology and Biomedicine*. Artech House, Bosten, MA, USA, 2006.

Ivo Anastácio, Bruno Martins, and Pável Calado. A Comparison of Different Approaches for Assigning Geographic Scopes to Documents. In *Proceedings of the 1st INForum-Simpósio de Informática*, pages 285–296. FCUL, 2009.

Gabor Angeli and Jakob Uszkoreit. Language-Independent Discriminative Parsing of Temporal Expressions. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL '13)*, pages 83–92. ACL, 2013.

Gabor Angeli, Christopher D. Manning, and Daniel Jurafsky. Parsing Time: Learning to Interpret Time Expressions. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT '12)*, pages 446–455. ACL, 2012.

Irem Arikan, Srikanta J. Bedathur, and Klaus Berberich. Time Will Tell: Leveraging Temporal Expressions in IR. In *Proceedings of the 2nd ACM International Conference on Web Search and Web Data Mining: Late Breaking-Results (WSDM '09)*. ACM, 2009.

Sitaram Asur, Srinivasan Parthasarathy, and Duygu Ucar. An Event-based Framework for Characterizing the Evolutionary Behavior of Interaction Graphs. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '07)*, pages 913–921. ACM, 2007.

Lars Backstrom, Jon Kleinberg, Ravi Kumar, and Jasmine Novak. Spatial Variation in Search Engine Queries. In *Proceedings of the 17th International Conference on World Wide Web (WWW '08)*, pages 357–366. ACM, 2008.

Ricardo Baeza-Yates. Searching the Future. In *Proceedings of the ACM SIGIR 2005 Workshop on Mathematical/Formal Methods in Information Retrieval (MFIR '05)*. ACM, 2005.

Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999.

Krisztian Balog, Arjen P. de Vries, Pavel Serdyukov, and Ji-Rong Wen. The First International Workshop on Entity-oriented Search (EOS). *ACM SIGIR Forum*, 45(2):43–50, 2012.

Valentina Bartalesi Lenzi and Rachele Sprugnoli. Evalita 2007: Description and Results of the TERN Task. *Intelligenza Artificiale*, 4(2):55–57, 2007.

David S. Batista, Mário J. Silva, Francisco M. Couto, and Bibek Behera. Geographic Signatures for Semantic Retrieval. In *Proceedings of the 6th Workshop on Geographic Information Retrieval (GIR '10)*, pages 19:1–19:8. ACM, 2010.

Michael Beaney, editor. *The Oxford Handbook of The History of Analytic Philosophy*. Oxford University Press, Oxford, UK, 2013.

Hila Becker, Mor Naaman, and Luis Gravano. Learning Similarity Metrics for Event Identification in Social Media. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining (WSDM '10)*, pages 291–300. ACM, 2010.

Steven M. Beitzel, Eric C. Jensen, Abdur Chowdhury, David Grossman, and Ophir Frieder. Hourly Analysis of a Very Large Topically Categorized Web Query Log. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '04)*, pages 321–328. ACM, 2004.

Klaus Berberich, Srikanta J. Bedathur, Thomas Neumann, and Gerhard Weikum. A Time Machine for Text Search. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*, pages 519–526. ACM, 2007.

Klaus Berberich, Srikanta J. Bedathur, Omar Alonso, and Gerhard Weikum. A Language Modeling Approach for Temporal Information Needs. In *Proceedings of the 32nd European Conference on Advances in Information Retrieval (ECIR '10)*, pages 13–25. Springer, 2010.

Steven Bethard. A Synchronous Context Free Grammar for Time Normalization. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP '13)*, pages 821–826. ACL, 2013a.

Steven Bethard. ClearTK-TimeML: A Minimalist Approach to TempEval 2013. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval '13)*, pages 10–14. ACL, 2013b.

Douglas Biber. *Variation across Speech and Writing*. Cambridge University Press, Cambridge, UK, 1988.

Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. O'Reilly, Sebastol, CA, USA, 2009.

André Bittar, Pascal Amsili, Pascal Denis, and Laurence Danlos. French TimeBank: an ISO-TimeML Annotated Reference Corpus. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL '11)*, pages 130–134. ACL, 2011.

Thomas Bögel, Jannik Strötgen, and Michael Gertz. Computational Narratology: Extracting Tense Clusters from Narrative Texts. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC '14)*, pages 950–955. ELRA, 2014.

Branimir Boguraev and Rie Kubota Ando. TimeBank-Driven TimeML Analysis. In Graham Katz, James Pustejovsky, and Frank Schilder, editors, *Annotating, Extracting and Reasoning about Time and Events*, number 05151 in Dagstuhl Seminar Proceedings, 2005.

Thorsten Brants and Reinhard Stolle. Finding Similar Documents in Document Collections. In *Proceedings on the LREC-2002 Workshop on Using Semantics for Information Retrieval and Filtering*. ELRA, 2002.

Sergey Brin and Lawrence Page. The Anatomy of a Large-scale Hypertextual Web Search Engine. In *Proceedings of the 7th International Conference on World Wide Web (WWW '98)*, pages 107–117. Elsevier, 1998.

Razvan C. Bunescu and Marius Pasca. Using Encyclopedic Knowledge for Named Entity Disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL '06)*, pages 9–16. ACL, 2006.

Tyler Burge. Gottlob Frege: Some Forms of Influence. In Michael Beaney, editor, *The Oxford Handbook of The History of Analytic Philosophy*, chapter 10, pages 355–382. Oxford University Press, Oxford, UK, 2013.

Stephan Busemann, Thierry Declerck, Abdel K. Diagne, Luca Dini, Judith Klein, and Sven Schmeier. Natural Language Dialogue Service for Appointment Scheduling Agents. In *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLC '97)*, pages 25–32. ACL, 1997.

Orkut Buyukkokten, Junghoo Cho, Hector Garcia-Molina, Luis Gravano, and Narayanan Shivakumar. Exploiting Geographical Location Information of Web Pages. In *Proceedings of the ACM SIGMOD Workshop on the Web and Databases (WebDB '99)*, pages 91–96. ACM, 1999.

Ricardo Campos, Alípio M. Jorge, and Gaël Dias. Using Web Snippets and Query-logs to Measure Implicit Temporal Intents in Queries. In *Proceedings of the SIGIR 2011 Workshop on Query Representation and Understanding (QRU '11)*, pages 13–16. ACM, 2011.

Ricardo Campos, Gaël Dias, Alípio M. Jorge, and Célia Nunes. GTE: A Distributional Second-order Co-occurrence Approach to Improve the Identification of Top Relevant Dates in Web Snippets. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*, pages 2035–2039. ACM, 2012.

Ricardo Campos, Gaël Dias, Alípio M. Jorge, and Adam Jatowt. Survey of Temporal Information Retrieval and Related Applications. *ACM Computing Surveys*, 47(2):15:1–15:41, 2014.

Nuno Cardoso and Mário J. Silva. A GIR Architecture with Semantic-flavored Query Reformulation. In *Proceedings of the 6th Workshop on Geographic Information Retrieval (GIR '10)*, pages 8:1–8:7. ACM, 2010a.

Nuno Cardoso and Mário J. Silva. Experiments with Semantic-flavored Query Reformulation of Geo-Temporal Queries. In *Proceedings of the 8th NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, pages 173–180. NII, 2010b.

Roberto Casati and Achille Varzi. Events. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Spring 2010 edition, 2010. URL `http://plato.stanford.edu/archives/spr2010/entries/events/` [last accessed June 10, 2014].

Tommaso Caselli, Felice dell'Orletta, and Irina Prodanof. TETI: a TimeML Compliant TimEx Tagger for Italian. In *Proceedings of the International Multiconference on Computer Science and Information Technology (IMCSIT '09)*, pages 185–192. IEEE, 2009.

Tommaso Caselli, Valentina Bartalesi Lenzi, Rachele Sprugnoli, Emanuele Pianta, and Irina Prodanof. Annotating Events, Temporal Expressions and Relations in Italian: the It-TimeML Experience for the Ita-TimeBank. In *Proceedings of the 5th Linguistic Annotation Workshop (LAW '11)*, pages 143–151. ACL, 2011.

William B. Cavnar and John M. Trenkle. N-Gram-Based Text Categorization. In *In Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval (SDAIR '94)*, pages 161–175. UNLV, 1994.

Nate Chambers. NavyTime: Event and Time Ordering from Raw Text. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval '13)*, pages 73–77. ACL, 2013.

Angel X. Chang and Christopher D. Manning. SUTime: A Library for Recognizing and Normalizing Time Expressions. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC '12)*, pages 3735–3740. ELRA, 2012.

Angel X. Chang and Christopher D. Manning. SUTime: Evaluation in TempEval-3. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval '13)*, pages 78–82. ACL, 2013.

Tao Chen and Min-Yen Kan. Creating a Live, Public Short Message Service Corpus: the NUS SMS Corpus. *Language Resources and Evaluation*, 47(2):299–335, 2013.

Yih-Farn R. Chen, Giuseppe Di Fabbrizio, David Gibbon, Serban Jora, Bernard Renger, and Bin Wei. Geotracker: Geospatial and Temporal RSS Navigation. In *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*, pages 41–50. ACM, 2007.

Zhiyuan Cheng, James Caverlee, and Kyumin Lee. You Are Where You Tweet: A Content-based Approach to Geo-locating Twitter Users. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM '10)*, pages 759–768. ACM, 2010.

Hung Chim and Xiaotie Deng. Efficient Phrase-based Document Similarity for Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 20(9):1217–1229, 2008.

Nancy A. Chinchor. Overview of MUC-7/MET-2. In *Proceedings of the 7th Message Understanding Conference (MUC '97)*, pages 1–11. Morgan Kaufmann, 1997.

Alexander Clark, Chris Fox, and Shalom Lappin. Introduction. In Alexander Clark, Chris Fox, and Shalom Lappin, editors, *The Handbook of Computational Linguistics and Natural Language Processing*, pages 1–8. Wiley-Blackwell, Oxford, UK, 2010a.

Alexander Clark, Chris Fox, and Shalom Lappin, editors. *The Handbook of Computational Linguistics and Natural Language Processing*. Wiley-Blackwell, Oxford, UK, 2010b.

Anthony G. Cohn, Brandon Bennett, John Gooday, and Nicholas M. Gotts. Qualitative Spatial Representation and Reasoning with the Region Connection Calculus. *Geoinformatica*, 1(3):275–316, 1997.

Francisco Costa and António Branco. TimeBankPT: A TimeML Annotated Corpus of Portuguese. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC '12)*, pages 3727–3734. ELRA, 2012.

Silviu Cucerzan. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL '07)*, pages 708–716. ACL, 2007.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Niraj Aswani, Ian Roberts, Genevieve Gorrell, Adam Funk, Angus Roberts, Danica Damljanovic, Thomas Heitz, Mark A. Greenwood, Horacio Saggion, Johann Petrak, Yaoyong Li, and Wim Peters. *Text Processing with GATE*. Gateway Press, Murphys, CA, USA, 2011.

Na Dai, Milad Shokouhi, and Brian D. Davison. Learning to Rank for Freshness and Relevance. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*, pages 95–104. ACM, 2011.

Wisam Dakka, Luis Gravano, and Panagiotis G. Ipeirotis. Answering General Time Sensitive Queries. In *Proceedings of the 17th ACM International Conference on Information and Knowledge Management (CIKM '08)*, pages 1437–1438. ACM, 2008.

Wisam Dakka, Luis Gravano, and Panagiotis G. Ipeirotis. Answering General Time-Sensitive Queries. *IEEE Transactions on Knowledge and Data Engineering*, 24(2):220–235, 2012.

Donald Davidson. The Individuation of Events. In Nicholas Rescher, editor, *Essays in Honor of Carl G. Hempel*. Springer, 1969. Reprinted in Donald Davidson, *Essays on Actions and Events*, essay 8, pages 163–180, Oxford University Press, Oxford, UK, 2nd edition, 2002.

Donald Davidson. Reply to Quine on Events. In *Actions and Events: Perspectives on the Philosophy of Donald Davidson*. Oxford Basil Blackwell, 1985. Reprinted in Donald Davidson, *Essays on Actions and Events*, appendix B, pages 305–311, Oxford University Press, Oxford, UK, 2nd edition, 2002.

Donald Davidson. *Essays on Action and Events*. Oxford University Press, Oxford, UK, 2nd edition, 2002.

Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. Generating Typed Dependency Parses from Phrase Structure Parses. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC '06)*, pages 449–454. ELRA, 2006.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6): 391–407, 1990.

Ludovic Denoyer and Patrick Gallinari. The Wikipedia XML Corpus. *ACM SIGIR Forum*, 40(1):64–69, 2006.

Leon Derczynski. *Determining the Types of Temporal Relations in Discourse*. PhD thesis, University of Sheffield, 2013.

Leon Derczynski and Robert Gaizauskas. USFD2: Annotating Temporal Expressions and TLINKs for TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval '10)*, pages 337–340. ACL, 2010.

Leon Derczynski, Hector Llorens, and Estela Saquete. Massively Increasing TIMEX3 Resources: A Transduction Approach. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC '12)*, pages 3754–3761. ELRA, 2012.

Gaël Dias, Ricardo Campos, and Alípio M. Jorge. Future Retrieval: What Does the Future Talk About? In *Proceedings of the SIGIR 2011 Workshop on Enriching Information Retrieval (ENIR '11)*, 2011.

Junyan Ding, Luis Gravano, and Narayanan Shivakumar. Computing Geographical Scopes of Web Resources. In *Proceedings of the 26th International Conference on Very Large Data Bases (VLDB '00)*, pages 545–556. Morgan Kaufmann, 2000.

George R. Doddington, Alexis Mitchell, Mark A. Przybocki, Lance A. Ramshaw, Stephanie Strassel, and Ralph M. Weischedel. The Automatic Content Extraction (ACE) Program - Tasks, Data, and Evaluation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC '06)*, pages 837–840. ELRA, 2004.

Michael Dummett. The Place of Philosophy in European Culture. *European Journal of Analytic Philosophy*, 3(1):21–30, 2007.

Miles Efron and Gene Golovchinsky. Estimation Methods for Ranking Recent Information. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*, pages 495–504. ACM, 2011.

Ali Farghaly and Khaled Shaalan. Arabic Natural Language Processing: Challenges and Solutions. *ACM Transactions on Asian Language Information Processing (TALIP)*, 8(4):14:1–14:22, 2009.

Manaal Faruqui and Sebastian Padó. Training and Evaluating a German Named Entity Recognizer with Semantic Generalization. In *Proceedings of the 10th Conference on Natural Language Processing (KONVENS '10)*, pages 129–133. Universaar, 2010.

Andreas Fay. Design and Implementation of a Temporal Query Language and Reranking Model. Student project, Heidelberg Universtiy, 2011.

Ronen Feldman and James Sanger. *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, New York, NY, USA, 2007.

Lisa Ferro, Inderjeet Mani, Beth Sundheim, and George Wilson. TIDES Temporal Annotation Guidelines – Version 1.0.2. Technical report, The MITRE Corporation, 2001.

Lisa Ferro, Laurie Gerber, Janet Hitzeman, Elizabeth Lima, and Beth Sundheim. ACE Time Normalization (TERN) 2004 English Training Data v 1.0. Linguistic Data Consortium (LDC), Philadelphia, PA, USA, 2005a. URL `https://catalog.ldc.upenn.edu/LDC2005T07` [last accessed October 10, 2014].

Lisa Ferro, Laurie Gerber, Inderjeet Mani, Beth Sundheim, and George Wilson. TIDES 2005 Standard for the Annotation of Temporal Expressions. Technical report, The MITRE Corporation, 2005b.

Lisa Ferro, Laurie Gerber, Inderjeet Mani, Beth Sundheim, and George Wilson. ACE Time Normalization (TERN) 2004 English Evaluation Data v 1.0. Linguistic Data Consortium (LDC), Philadelphia, PA, USA, 2010. URL `https://catalog.ldc.upenn.edu/LDC2010T18` [last accessed October 10, 2014].

David A. Ferrucci and Adam Lally. Building an Example Application with the Unstructured Information Management Architecture. *IBM Systems Journal*, 43(3):455–475, 2004a.

David A. Ferrucci and Adam Lally. UIMA: an Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *Natural Language Engineering*, 10(3-4):327–348, 2004b.

Radu Florian, Hany Hassan, Abraham Ittycheriah, Hongyan Jing, Nanda Kambhatla, Xiaoqiang Luo, Nicolas Nicolov, and Salim Roukos. A Statistical Model for Multilingual Entity Detection and Tracking. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL '04)*, pages 1–8. ACL, 2004.

Corina Forascu and Dan Tufis. Romanian TimeBank: An Annotated Parallel Corpus for Temporal Information. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC '12)*, pages 3762–3766. ELRA, 2012.

Santo Fortunato. Community Detection in Graphs. *Physics Reports*, 486(3-5):75–174, 2010.

Oliver Fuchs. An Advanced Interface for Geographic Querying. Student project, Heidelberg Universtiy, 2014.

Katrin Fundel, Robert Küffner, and Ralf Zimmer. RelEx—Relation Extraction Using Dependency Parse Trees. *Bioinformatics*, 23(3):365–371, 2007.

Evgeniy Gabrilovich and Shaul Markovitch. Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence (IJCAI '07)*, pages 1606–1611. Morgan Kaufmann, 2007.

Volker Gaede and Oliver Günther. Multidimensional Access Methods. *ACM Computing Surveys*, 30(2): 170–231, 1998.

William A. Gale, Kenneth W. Church, and David Yarowsky. One Sense Per Discourse. In *Proceedings of the 4th DARPA Speech and Natural Language Workshop*, pages 233–237. ACL, 1992.

Fredric Gey, Ray R. Larson, Mark Sanderson, Hideo Joho, Paul Clough, and Vivien Petras. GeoCLEF: The CLEF 2005 Cross-Language Geographic Information Retrieval Track Overview. In *Proceedings of the 6th Workshop of the Cross-Language Evaluation Forum (CLEF '05)*, pages 908–919. Springer, 2006.

Fredric Gey, Ray R. Larson, Mark Sanderson, Kerstin Bischoff, Thomas Mandl, Christa Womser-Hacker, Diana Santos, Paulo Rocha, Giorgio M. Di Nunzio, and Nicola Ferro. GeoCLEF 2006: The CLEF 2006 Cross-Language Geographic Information Retrieval Track Overview. In *Proceedings of the 7th Workshop of the Cross-Language Evaluation Forum (CLEF '06)*, pages 852–876. Springer, 2007.

Fredric Gey, Ray R. Larson, Noriko Kando, Jorge Machado, and Tetsuya Sakai. NTCIR-GeoTime Overview: Evaluating Geographic and Temporal Search. In *Proceedings of the 8th NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, pages 147–153. NII, 2010.

Fredric Gey, Ray R. Larson, Jorge Machado, and Masaharu Yoshioka. NTCIR9-GeoTime Overview - Evaluating Geographic and Temporal Search: Round 2. In *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, pages 9–17. NII, 2011.

Fredric C. Gey, Ryan Shaw, Ray R. Larson, and Barry Pateman. Biography as Events in Time and Space. In *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS '08)*, pages 537–538. ACM, 2008.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL '11)*, pages 42–47. ACL, 2011.

Chung Heong Gooi and James Allan. Cross-Document Coreference on a Large Scale Corpus. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL '04)*, pages 9–16. ACL, 2004.

Luis Gravano, Vasileios Hatzivassiloglou, and Richard Lichtenstein. Categorizing Web Queries According to Geographical Locality. In *Proceedings of the 12th ACM International Conference on Information and Knowledge Management (CIKM '03)*, pages 325–333. ACM, 2003.

Ralph Grishman. Information Extraction. In Alexander Clark, Chris Fox, and Shalom Lappin, editors, *The Handbook of Computational Linguistics and Natural Language Processing*, chapter 18, pages 517–530. Wiley-Blackwell, Oxford, UK, 2010.

Ralph Grishman and Beth Sundheim. Design of the MUC-6 Evaluation. In *Proceedings of the 6th Message Understanding Conference (MUC '95)*, pages 1–11. Morgan Kaufmann, 1995.

Claire Grover, Richard Tobin, Beatrice Alex, and Kate Byrne. Edinburgh-LTG: TempEval-2 System Description. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval '10)*, pages 333–336. ACL, 2010.

Marta Guerrero Nieto and Roser Saurí. ModeS TimeBank 1.0. Linguistic Data Consortium (LDC), Philadelphia, PA, USA, 2012. URL `https://catalog.ldc.upenn.edu/LDC2012T01` [last accessed October 10, 2014].

Marta Guerrero Nieto, Roser Saurí, and Miguel A. Bernabe Poveda. ModeS TimeBank: A Modern Spanish TimeBank Corpus (ModeS TimeBank: Un Corpus TimeBank del Español Moderno). *Procesamiento del Lenguaje Natural*, 47(1):259 – 267, 2011.

James Gung and Jugal Kalita. Summarization of Historical Articles Using Temporal Event Clustering. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT '12)*, pages 631–635. ACL, 2012.

Iryna Gurevych, Max Mühlhäuser, Christof Müller, Jürgen Steimle, Markus Weimer, and Torsten Zesch. Darmstadt Knowledge Processing Repository Based on UIMA. In *Proceedings of the First Workshop on Unstructured Information Management Architecture at Biannual Conference of the German Society for Computational Linguistics and Language Technology*, 2007.

Kadri Hacioglu, Ying Chen, and Benjamin Douglas. Automatic Time Expression Labeling for English and Chinese Text. In *Proceedings of the 6th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing '05)*, pages 548–559. Springer, 2005.

Xianpei Han, Le Sun, and Jun Zhao. Collective Entity Linking in Web Text: A Graph-based Method. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*, pages 765–774. ACM, 2011.

Marti A. Hearst. Untangling Text Data Mining. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL '99)*, pages 3–10. ACL, 1999.

Andreas Heuer and Marc H. Scholl. Principles of Object-Oriented Query Languages. In *Proceedings der GI-Fachtagung für Datenbanksysteme in Büro, Technik und Wissenschaft (BTW '91)*, pages 178–197. Springer, 1991.

Linda L. Hill. *Georeferencing – The Geographic Associations of Information.* MIT Press, Cambridge, MA, USA, 2006.

Erhard Hinrichs. Temporal Anaphora in Discourses of English. *Linguistics and Philosophy*, 9(1):63–82, 1986.

Guoping Hu, Jingjing Liu, Hang Li, Yunbo Cao, Jian-Yun Nie, and Jianfeng Gao. A Supervised Learning Approach to Entity Search. In *Proceedings of the 3rd Asia Conference on Information Retrieval Technology (AIRS '06)*, pages 54–66. Springer, 2006.

Kalervo Järvelin and Jaana Kekäläinen. Cumulated Gain-based Evaluation of IR Techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

Adam Jatowt, Kensuke Kanazawa, Satoshi Oyama, and Katsumi Tanaka. Supporting Analysis of Future-related Information in News Archives and the Web. In *Proceedings of the 9th ACM/IEEE Joint Conference on Digital Libraries (JCDL '09)*, pages 115–124. ACM, 2009.

Adam Jatowt, Ching-Man Au Yeung, and Katsumi Tanaka. Estimating Document Focus Time. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM '13)*, pages 2273–2278. ACM, 2013.

Jing Jiang. Information Extraction from Text. In Charu C. Aggarwal and ChengXiang Zhai, editors, *Mining Text Data*, chapter 2, pages 11–41. Springer, New York, NY, USA, 2012.

Peiquan Jin, Jianlong Lian, Xujian Zhao, and Shouhong Wan. TISE: A Temporal Search Engine for Web Contents. In *Proceedings of the 2nd International Symposium on Intelligent Information Technology Application (IITA '08)*, pages 220–224. IEEE, 2008.

Prateek Jindal and Dan Roth. Extraction of Events and Temporal Expressions from Clinical Narratives. *Journal of Biomedical Information*, 46 Suppl:S13–S19, 2013.

Christopher B. Jones and Ross Purves, editors. *Proceedings of the 2nd Workshop on Geographic Information Retrieval (GIR '05)*, New York, NY, USA, 2005. ACM.

Christopher B. Jones and Ross Purves. GIR'05 2005 ACM Workshop on Geographical Information Retrieval. *ACM SIGIR Forum*, 40(1):34–37, 2006.

Christopher B. Jones and Ross Purves, editors. *Proceedings of the 5th Workshop on Geographic Information Retrieval (GIR '08)*, New York, NY, USA, 2008. ACM.

Christopher B. Jones and Ross Purves, editors. *Proceedings of the 7th Workshop on Geographic Information Retrieval (GIR '13)*, New York, NY, USA, 2013. ACM.

Christopher B. Jones, Ross Purves, Anne Ruas, Mark Sanderson, Monika Sester, Marc J. van Kreveld, and Robert Weibel. Spatial Information Retrieval and Geographical Ontologies an Overview of the SPIRIT Project. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '02)*, pages 387–388. ACM, 2002.

Rosie Jones and Fernando Diaz. Temporal Profiles of Queries. *ACM Transactions on Information Systems*, 25(3):14:1–14:31, 2007.

Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Prentice Hall, Upper Saddle River, NJ, USA, 2nd edition, 2008.

Nattiya Kanhabua and Wolfgang Nejdl. Understanding the Diversity of Tweets in the Time of Outbreaks. In *Proceedings of the 22nd International Conference on World Wide Web (WWW '13)*, pages 1335–1342. ACM, 2013.

Nattiya Kanhabua and Kjetil Nørvåg. Determining Time of Queries for Re-ranking Search Results. In *Proceedings of the 14th European Conference on Research and Advanced Technology for Digital Libraries (ECDL '10)*, pages 261–272. Springer, 2010.

Nattiya Kanhabua and Kjetil Nørvåg. Learning to Rank Search Results for Time-sensitive Queries. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*, pages 2463–2466. ACM, 2012.

Nattiya Kanhabua, Sara Romano, Avaré Stewart, and Wolfgang Nejdl. Supporting Temporal Analytics for Health-related Events in Microblogs. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*, pages 2686–2688. ACM, 2012.

Christian Kapp, Jannik Strötgen, and Michael Gertz. EvenPers: Event-based Person Exploration and Correlation. In *Proceedings der 15. GI-Fachtagung für Datenbanksysteme in Business, Technologie und Web (BTW '13)*, pages 519–522. Springer, 2013.

Manuel Kaufmann. Modellierung und Analyse heuristischer und linguistischer Methoden zur verbesserten Eventextraktion. Bachelor's thesis, Heidelberg Universtiy, 2012.

Britta Keller, Jannik Strötgen, and Michael Gertz. Event-centric Document Similarity for Biomedical Literature. In *Proceedings of the 5th International Symposium on Semantic Mining in Biomedicine (SMBM '12)*, pages 72–79. ZORA, 2012.

Mahboob A. Khalid, Valentin Jijkoun, and Maarten De Rijke. The Impact of Named Entity Normalization on Information Retrieval for Question Answering. In *Proceedings of the 30th European Conference on Advances in Information Retrieval (ECIR '08)*, pages 705–710. Springer, 2008.

Houda Khrouf and Raphaël Troncy. Hybrid Event Recommendation Using Linked Data and User Diversity. In *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys '13)*, pages 185–192. ACM, 2013.

Jaegwon Kim. Causation, Nomic Subsumption, and the Concept of Event. *The Journal of Philosophy*, 70 (8):217–236, 1973.

Jin-Dong Kim, Tomoko Ohta, Yuka Teteisi, and Jun'ichi Tsujii. GENIA Corpus Manual – Encoding Schemes for the Corpus and Annotation. Technical report, Tsujii Laboratory, University of Tokyo, 2006.

Jin-Dong Kim, Tomoko Ohta, and Jun'ichi Tsujii. Corpus Annotation for Mining Biomedical Events from Literature. *BMC Bioinformatics*, 9(1):10, 2008.

Jin-Dong Kim, Tomoko Ohta, Sampo Pyysalo, Yoshinobu Kano, and Jun'ichi Tsujii. Overview of BioNLP'09 Shared Task on Event Extraction. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task (BioNLP '09)*, pages 1–9. ACL, 2009.

Oleksandr Kolomiyets and Marie-Francine Moens. Meeting TempEval-2: Shallow Approach for Temporal Tagger. In *Proceedings of the NAACL-HLT Workshop on Semantic Evaluations: Recent Achievements and Future Directions (SEW '09)*, pages 52–57. ACL, 2009.

Oleksandr Kolomiyets and Marie-Francine Moens. KUL: Recognition and Normalization of Temporal Expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval '10)*, pages 325–328. ACL, 2010.

Martin Krallinger, Florian Leitner, Carlos Rodriguez-Penagos, and Alfonso Valencia. Overview of the Protein-Protein Interaction Annotation Extraction Task of BioCreative II. *Genome Biology*, 9(Suppl 2): S4, 2008.

Saul A. Kripke. *Naming and Necessity*. Harvard University Press, Cambridge, MA, USA, 1980.

Bhaskar Krishnamachari, Deborah Estrin, and Stephen B. Wicker. The Impact of Data Aggregation in Wireless Sensor Networks. In *Proceedings of the 22nd International Conference on Distributed Computing Systems (ICDCSW '02)*, pages 575–578. IEEE, 2002.

Anagha Kulkarni, Jaime Teevan, Krysta M. Svore, and Susan T. Dumais. Understanding Temporal Query Dynamics. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM '11)*, pages 167–176. ACM, 2011.

Chandan Kumar, Wilko Heuten, and Susanne Boll. Geographical Queries Beyond Conventional Boundaries: Regional Search and Exploration. In *Proceedings of the 7th International Workshop on Geographic Information Retrieval (GIR '13)*, pages 84–85. ACM, 2013.

Anup Kumar Kolya, Asif Ekbal, and Sivaji Bandyopadhyay. JU_CSE_TEMP: A First Step towards Evaluating Events, Time Expressions and Temporal Relations. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval '10)*, pages 345–350. ACL, 2010.

Praveen Lakkaraju, Susan Gauch, and Mirco Speretta. Document Similarity based on Concept Tree Distance. In *Proceedings of the 19th ACM Conference on Hypertext and Hypermedia (HT '08)*, pages 127–132. ACM, 2008.

Ivan Laptev. On Space-Time Interest Points. *International Journal on Computer Vision*, 64(2-3):107–123, 2005.

Ray R. Larson. Geographic Information Retrieval and Digital Libraries. In *Proceedings of the 13th European Conference on Research and Advanced Technology for Digital Libraries (ECDL '09)*, pages 461–464. Springer, 2009a.

Ray R. Larson. Cheshire at GeoCLEF 2008: Text and Fusion Approaches for GIR. In *Proceedings of the 9th Workshop of the Cross-Language Evaluation Forum (CLEF '08)*, pages 830–837. Springer, 2009b.

Ray R. Larson and Patricia Frontiera. Geographic Information Retrieval (GIR) Ranking Methods for Digital Libraries. In *Proceedings of the 4th ACM/IEEE Joint Conference on Digital Libraries (JCDL '04)*, page 415. ACM, 2004.

Hady W. Lauw, Ee-Peng Lim, HweeHwa Pang, and Teck-Tim Tan. Social Network Discovery by Mining Spatio-Temporal Events. *Computational & Mathematical Organization Theory*, 11(2):97–118, 2005.

David Y. W. Lee. Genres, Registers, Text Types, Domains and Styles: Clarifying the Concepts and Navigating a Path through the BNC Jungle. *Language Learning & Technology*, 5(3):37–72, 2001.

Michael D. Lee, Brandon Pincombe, and Matthew Welsh. An Empirical Evaluation of Models of Text Document Similarity. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society (CogSci '05)*, pages 1254–1259. Erlbaum, 2005.

Jochen L. Leidner. *Toponym Resolution in Text: Annotation, Evaluation and Applications of Spatial Grounding of Place Names*. PhD thesis, University of Edinburgh, 2007.

Jochen L. Leidner and Michael D. Lieberman. Detecting Geographical References in the Form of Place Names and Associated Spatial Natural Language. *SIGSPATIAL Special*, 3(2):5–11, 2011.

Jochen L. Leidner, Gail Sinclair, and Bonnie Webber. Grounding Spatial Named Entities for Information Extraction and Question Answering. In *Proceedings of the HLT-NAACL 2003 Workshop on Analysis of Geographic References*, pages 31–38. ACL, 2003.

Ernie Lepore and Kirk Ludwig. *A Companion to Donald Davidson*. Wiley-Blackwell, New York, NY, USA, 2013.

Johannes Leveling and Sven Hartrumpf. On Metonymy Recognition for Geographic Information Retrieval. *International Journal of Geographical Information Science*, 22(3):289–299, 2008.

Hui Li, Jannik Strötgen, Julian Zell, and Michael Gertz. Chinese Temporal Tagging with HeidelTime. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL '14)*, pages 133–137. ACL, 2014.

Huifeng Li, Rohini K. Srihari, Cheng Niu, and Wei Li. Location Normalization for Information Extraction. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING '02)*, pages 549–555. ACL, 2002.

Xiaoyan Li and W. Bruce Croft. Time-based Language Models. In *Proceedings of the 12th ACM International Conference on Information and Knowledge Management (CIKM '03)*, pages 469–475. ACM, 2003.

Zhisheng Li, Ken C. K. Lee, Baihua Zheng, Wang-Chien Lee, Dik Lee, and Xufasash Wang. IR-Tree: An Efficient Index for Geographic Document Search. *IEEE Transactions on Knowledge and Data Engineering*, 23(4):585–599, 2011.

Shasha Liao and Ralph Grishman. Using Document Level Cross-event Inference to Improve Event Extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL '10)*, pages 789–797. ACL, 2010.

Michael D. Lieberman, Hanan Samet, Jagan Sankaranarayanan, and Jon Sperling. STEWARD: Architecture of a Spatio-textual Search Engine. In *Proceedings of the 15th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS '07)*, pages 186–193. ACM, 2007.

Michael D. Lieberman, Hanan Samet, and Jagan Sankaranarayanan. Geotagging with Local Lexicons to Build Indexes for Textually-specified Spatial Data. In *Proceedings of the 26th International Conference on Data Engineering (ICDE '10)*, pages 201–212. IEEE, 2010.

Tobias Limpert. Verbesserung der spatio-temporal Event Extraktion und ihrer Kontextinformation durch Relationsextraktionsmethoden. Bachelor's thesis, Heidelberg Universtiy, 2013.

Hector Llorens, Estela Saquete, and Borja Navarro. TIPSem (English and Spanish): Evaluating CRFs and semantic roles in TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval '10)*, pages 284–291. ACL, 2010.

Hector Llorens, Leon Derczynski, Robert Gaizauskas, and Estela Saquete. TIMEN: An Open Temporal Expression Normalisation Resource. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC '12)*, pages 3044–3051. ELRA, 2012a.

Hector Llorens, Naushad UzZaman, and James F. Allen. Merging Temporal Annotations. In *Proceedings of the 19th International Symposium on Temporal Representation and Reasoning (TIME '12)*, pages 107–113. IEEE, 2012b.

Jorge Machado, Bruno Martins, and José Borbinha. LGTE: Lucene Extensions for Geo-Temporal Information Retrieval. In *Workshop on Geographic Information on the Internet (GIIW '09)*, 2009.

Jorge Machado, José Borbinha, and Bruno Martins. Experiments with Geo-Temporal Expressions Filtering and Query Expansion at Document and Phrase Context Resolution. In *Proceedings of the 8th NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, pages 159–166. NII, 2010.

Jorge Machado, José Borbinha, and Bruno Martins. Geo-Temporal Retrieval Filtering versus Answer Resolution Using Wikipedia. In *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, pages 38–47. NII, 2011.

Walid Magdy, Kareem Darwish, Ossama Emam, and Hany Hassan. Arabic Cross-document Person Name Normalization. In *Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources (Semitic '07)*, pages 25–32. ACL, 2007.

Bernardo Magnini, Emanuele Pianta, Christian Girardi, Matteo Negri, Lorenza Romano, Manuela Speranza, Valentina Bartalesi Lenzi, and Rachele Sprugnoli. I-CAB: the Italian Content Annotation Bank. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC '06)*, pages 963–968. ELRA, 2006.

Juha Makkonen, Helena Ahonen-myka, and Marko Salmenkivi. Topic Detection and Tracking with Spatio-Temporal Evidence. In *Proceedings of the 25th European Conference on Advances in Information Retrieval (ECIR '03)*, pages 251–265. Springer, 2003.

Thomas Mandl, Fredric Gey, Giorgio M. Di Nunzio, Nicola Ferro, Ray R. Larson, Mark Sanderson, Diana Santos, Christa Womser-Hacker, and Xing Xie. GeoCLEF 2007: The CLEF 2007 Cross-Language Geographic Information Retrieval Track Overview. In *Proceedings of the 8th Workshop of the Cross-Language Evaluation Forum (CLEF '07)*, pages 745–772. Springer, 2008.

Thomas Mandl, Paula Carvalho, Giorgio Maria Di Nunzio, Fredric Gey, Ray R. Larson, Diana Santos, and Christa Womser-Hacker. GeoCLEF 2008: The CLEF 2008 Cross-Language Geographic Information Retrieval Track Overview. In *Proceedings of the 9th Workshop of the Cross-Language Evaluation Forum (CLEF '08)*, pages 808–821. Springer, 2009.

Inderjeet Mani and George Wilson. Robust Temporal Processing of News. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL '00)*, pages 69–76. ACL, 2000a.

Inderjeet Mani and George Wilson. Temporal Granularity and Temporal Tagging of Text. In *Proceedings of the AAAI-2000 Workshop on Spatial and Temporal Granularity*, pages 71–73. AAAI, 2000b.

Inderjeet Mani, Janet Hitzeman, Justin Richer, Dave Harris, Rob Quimby, and Ben Wellner. SpatialML: Annotation Scheme, Corpora, and Tools. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC '08)*, pages 410–415. ELRA, 2008.

Edimar Manica, Carina F. Dorneles, and Renata Renata Galante. Handling Temporal Information in Web Search Engines. *SIGMOD Record*, 41(3):15–23, 2012.

Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, MA, USA, 6th edition, 2003.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

Alexander Markowetz, Yen yu Chen, Torsten Suel, Xiaohui Long, and Bernhard Seeger. Design and Implementation of a Geographic Search Engine. In *Proceedings of the 8th International Workshop on the Web and Databases (WebDB '05)*, pages 19–24, 2005.

Bruno Martins and Pável Calado. Learning to Rank for Geographic Information Retrieval. In *Proceedings of the 6th Workshop on Geographic Information Retrieval (GIR '10)*, pages 21:1–21:8. ACM, 2010.

Bruno Martins, Mário J. Silva, and Leonardo Andrade. Indexing and Ranking in Geo-IR Systems. In *Proceedings of the 2nd Workshop on Geographic Information Retrieval (GIR '05)*, pages 31–34. ACM, 2005.

Bruno Martins, Hugo Manguinhas, and José Borbinha. Extracting and Exploring the Geo-Temporal Semantics of Textual Resources. In *Proceedings of the 2008 IEEE International Conference on Semantic Computing (ICSC '08)*, pages 1–9. IEEE, 2008.

Felix Mata and Christophe Claramunt. GeoST: Geographic, Thematic and Temporal Information Retrieval from Heterogeneous Web Data Sources. In *Proceedings of the 10th International Symposium on Web and Wireless Geographical Information Systems (W2GIS '11)*, pages 5–20. Springer, 2011.

Michael Matthews, Pancho Tolchinsky, Roi Blanco, Jordi Atserias, Peter Mika, and Hugo Zaragoza. Searching through Time in the New York Times. In *Proceedings of the 4th Workshop on Human-Computer Interaction and Information Retrieval (HCIR '10)*, pages 41–44. ACM, 2010.

Pawel Mazur. *Broad-coverage Rule-based Processing of Temporal Expressions*. PhD thesis, Macquarie University and Wroclaw University of Technology, 2012.

Pawel Mazur and Robert Dale. The DANTE Temporal Expression Tagger. In *Proceedings of the 3rd Language and Technology Conference (LTC '09)*, pages 245–257. Springer, 2009.

Pawel Mazur and Robert Dale. WikiWars: A New Corpus for Research on Temporal Expressions. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP '10)*, pages 913–922. ACL, 2010.

Pawel Mazur and Robert Dale. LTIMEX: Representing the Local Semantics of Temporal Expressions. In *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS '11)*, pages 201–208. IEEE, 2011.

Sharad Mehrotra, Carter Butts, Dmitri V. Kalashnikov, Nalini Venkatasubramanian, Ramesh Rao, Ganz Chockalingam, Ron T. Eguchi, Beverly Adams, and Charles Huyck. Project Rescue: Challenges in Responding to the Unexpected. In *Proceedings of 16th Annual Symposium on Electronic Imaging Science and Technology*, pages 179–192. SPIE, 2004.

Donald Metzler, Rosie Jones, Fuchun Peng, and Ruiqiang Zhang. Improving Search Relevance for Implicitly Temporal Queries. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09)*, pages 700–701. ACM, 2009.

Wu Mingli, Li Wenjie, Lu Qin, and Li Baoli. CTEMP: A Chinese Temporal Parser for Extracting and Normalizing Temporal Information. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP '05)*, pages 694–706. Springer, 2005.

Véronique Moriceau and Xavier Tannier. French Resources for Extraction and Normalization of Temporal Expressions with HeidelTime. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC '14)*, pages 3239–3243. ELRA, 2014.

David Mountain and Andrew MacFarlane. Geographic Information Retrieval in a Mobile Environment: Evaluating the Needs of Mobile Individuals. *Journal of Information Science*, 33(5):515–530, 2007.

David Nadeau and Satoshi Sekine. A Survey of Named Entity Recognition and Classification. *Linguisticae Investigationes*, 30(1):3–26, 2007.

Matteo Negri. Dealing with Italian Temporal Expressions: The ITA-CHRONOS System. *Intelligenza Artificiale*, 4(2):58–59, 2007.

Matteo Negri and Luca Marseglia. Recognition and Normalization of Time Expressions: ITC-irst at TERN 2004. Technical report, ITC-irst Trento, 2004.

Matteo Negri, Estela Saquete, Patricio Martínez-Barco, and Rafael Muñoz. Evaluating Knowledge-based Approaches to the Multilingual Extension of a Temporal Expression Normalizer. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events (ARTE '06)*, pages 30–37. ACL, 2006.

Cam-Tu Nguyen, Xuan-Hieu Phan, and Thu-Trang Nguyen. JVnTextPro: a Tool to Process Vietnamese Texts. Version 2.0, 2010. URL `http://jvntextpro.sourceforge.net/` [last accessed April 8,2014].

Dinh-Hoa Nguyen. *Vietnamese*. John Benjamins Publishing Company, Amsterdam, Netherlands, 1997.

Sérgio Nunes, Cristina Ribeiro, and Gabriel David. Use of Temporal Expressions in Web Search. In *Proceedings of the 30th European Conference on Advances in Information Retrieval (ECIR '08)*, pages 580–584. Springer, 2008.

Damien Palacio, Guillaume Cabanac, Christian Sallaberry, and Gilles Hubert. Measuring Effectiveness of Geographic IR Systems in Digital Libraries: Evaluation Framework and Case Study. In *Proceedings of the 14th European Conference on Research and Advanced Technology for Digital Libraries (ECDL '10)*, pages 340–351. Springer, 2010.

Martha Palmer and Nianwen Xue. Linguistic Annotation. In Alexander Clark, Chris Fox, and Shalom Lappin, editors, *The Handbook of Computational Linguistics and Natural Language Processing*, chapter 10, pages 238–270. Wiley-Blackwell, Oxford, UK, 2010.

Jong C. Park and Jung-jae Kim. Named Entity Recognition. In Sophia Ananiadou and John McNaught, editors, *Text Mining for Biology and Biomedicine*, chapter 6, pages 121–142. Artech House, Bosten, MA, USA, 2006.

Marius Pasca. Towards Temporal Web Search. In *Proceedings of the 2008 ACM Symposium on Applied Computing (SAC '08)*, pages 1117–1121. ACM, 2008.

Dieter Pfoser, Alexandros Efentakis, Thanasis Hadzilacos, Sophia Karagiorgou, and Giorgos Vasiliou. Providing Universal Access to History Textbooks: A Modified GIS Case. In *Proceedings of the 9th International Symposium on Web and Wireless Geographical Information Systems (W2GIS '09)*, pages 87–102. Springer, 2009.

Marcel Puchol-Blasco, Estela Saquete, and Patricio Martínez-Barco. Multilingual Extension of Temporal Expression Recognition Using Parallel Corpora. In *Proceedings of the 14th International Symposium on Temporal Representation and Reasoning (TIME '07)*, pages 175–180. IEEE, 2007.

Ross Purves and Christopher B. Jones, editors. *Proceedings of the 4th Workshop on Geographic Information Retrieval (GIR '07)*, New York, NY, USA, 2007. ACM.

Ross Purves, Paul Clough, and Christopher B. Jones, editors. *Proceedings of the 6th Workshop on Geographic Information Retrieval (GIR '10)*, New York, NY, USA, 2010. ACM.

Ross S. Purves, Paul Clough, Christopher B. Jones, Avi Arampatzis, Benedicte Bucher, David Finch, Gaihua Fu, Hideo Joho, Awase Khirni Syed, Subodh Vaid, and Bisheng Yang. The Design and Implementation of SPIRIT: a Spatially Aware Search Engine for Information Retrieval on the Internet. *International Journal of Geographical Information Science*, 21(7):717–745, 2007.

James Pustejovsky, José M. Castaño, Robert Ingria, Roser Saurí, Robert J. Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir R. Radev. TimeML: Robust Specification of Event and Temporal Expressions in Text. In *New Directions in Question Answering*, pages 28–34. AAAI, 2003a.

James Pustejovsky, Patrick Hanks, Roser Saurí, Andrew See, Robert J. Gaizauskas, Andrea Setzer, Dragomir R. Radev, Beth Sundheim, David Day, Lisa Ferro, and Marcia Lazo. The TIMEBANK Corpus. In *Proceedings of Corpus Linguistics 2003*, pages 647–656. UCREL, 2003b.

James Pustejovsky, Robert Knippen, Jessica Littman, and Roser Saurí. Temporal and Event Information in Natural Language Text. *Language Resources and Evaluation*, 39(2-3):123–164, 2005.

Yves Rasolofo and Jacques Savoy. Term proximity scoring for keyword-based retrieval systems. In *Proceedings of the 25th European Conference on Advances in Information Retrieval (ECIR '03)*, pages 207–218. Springer, 2003.

Philip Resnik and Jimmy Lin. Evaluation of NLP Systems. In Alexander Clark, Chris Fox, and Shalom Lappin, editors, *The Handbook of Computational Linguistics and Natural Language Processing*, chapter 11, pages 271–295. Wiley-Blackwell, Oxford, UK, 2010.

Philippe Rigaux, Michel Scholl, and Agnès Voisard. *Spatial Databases with Applications to GIS*. Morgan Kaufmann, San Francisco, CA, USA, 2002.

Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at TREC-3. In *Proceedings of the 3rd Text REtrieval Conference (TREC '94)*, pages 109–126. NIST, 1994.

Gunter Saake, Kai-Uwe Sattler, and Andreas Heuer. *Datenbanken – Konzepte und Sprachen*. MITP, Heidelberg, Germany, 4. Auflage, 2010.

Iman Saleh, Lamia Tounsi, and Josef van Genabith. ZamAn and Raqm: Extracting Temporal and Numerical Expressions in Arabic. In *Proceedings of the 7th Asia Conference on Information Retrieval Technology (AIRS '11)*, pages 562–573. Springer, 2011.

Jagan Sankaranarayanan, Hanan Samet, Benjamin E. Teitler, Michael D. Lieberman, and Jon Sperling. TwitterStand: News in Tweets. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS '09)*, pages 42–51. ACM, 2009.

Diana Santos and Luís Miguel Cabral. GikiCLEF: Expectations and Lessons Learned. In *Proceedings of the 10th Workshop of the Cross-Language Evaluation Forum (CLEF '09)*, pages 212–222. Springer, 2010.

Diana Santos and Nuno Cardoso. GikiP: Evaluating Geographical Answers from Wikipedia. In *Proceedings of the 5th Workshop on Geographic Information Retrieval (GIR '08)*, pages 59–60. ACM, 2008.

Diana Santos, Nuno Cardoso, Paula Carvalho, Iustin Dornescu, Sven Hartrumpf, Johannes Leveling, and Yvonne Skalban. Getting Geographical Answers from Wikipedia: the GikiP Pilot at CLEF. In *Working Notes for the CLEF 2008 Workshop (CLEF '08)*, 2008.

Diana Santos, Luís Miguel Cabral, Corina Forascu, Pamela Forner, Fredric Gey, Katrin Lamm, Thomas Mandl, Petya Osenova, Anselmo Peñas, Álvaro Rodrigo, Julia Schulz, Yvonne Skalban, and Erik Tjong Kim Sang. GikiCLEF: Crosscultural Issues in Multilingual Information Access. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC '10)*, pages 2346–2353. ELRA, 2010.

Estela Saquete. ID 392: TERSEO + T2T3 Transducer. A systems for Recognizing and Normalizing TIMEX3. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval '10)*, pages 317–320. ACL, 2010.

Estela Saquete and James Pustejovsky. Automatic Transformation from TIDES to TimeML Annotation. *Language Resources and Evaluation*, 45(4):495–523, 2011.

Estela Saquete, Patricio Martínez-Barco, and Rafael Muñoz. Evaluation of the Automatic Multilinguality for Time Expression Resolution. In *Proceedings of the 15th International Workshop on Database and Expert Systems Applications (DEXA '04)*, pages 25–30. IEEE, 2004.

Estela Saquete, Patricio Martínez-Barco, Rafael Muñoz, Matteo Negri, Manuela Speranza, and Rachele Sprugnoli. Multilingual Extension of a Temporal Expression Normalizer Using Annotated Corpora. In *Proceedings of the International Workshop on Cross-Language Knowledge Induction*, pages 1–8. ACL, 2006a.

Estela Saquete, Rafael Muñoz, and Patricio Martínez-Barco. Event Ordering using TERSEO System. *Data and Knowledge Engineering*, 58(1):70–89, 2006b.

Roser Saurí and Toni Badia. Spanish TimeBank 1.0. Linguistic Data Consortium (LDC), Philadelphia, PA, USA, 2012. URL `https://catalog.ldc.upenn.edu/LDC2012T12` [last accessed October 10, 2014].

Roser Saurí, Jessica Littman, Bob Knippen, Robert Gaizauskas, Andrea Setzer, and James Pustejovsky. TimeML Annotation Guidelines Version 1.2.1, January 2006. ULR `http://timeml.org/site/publications/timeMLdocs/annguide_1.2.1.pdf` [last accessed October 10, 2014].

Frank Schilder and Christopher Habel. From Temporal Expressions to Temporal Information: Semantic Tagging of News Messages. In *Proceedings of the ACL-2001 Workshop on Temporal and Spatial Information Processing (TASIP '01)*, pages 65–72. ACL, 2001.

Helmut Schmid. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of the International Conference on New Methods in Language Processing*, pages 44–49, 1994.

Susan Schneider. Events. In James Fieser and Bradley Dowden, editors, *The Internet Encyclopedia of Philosophy (IEP)*. 2014. URL `http://web.archive.org/web/20140605073753/http://www.iep.utm.edu/events/` [last accessed June 25, 2014].

Steven Schockaert and Martine De Cock. Neighborhood Restrictions in Geographic IR. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*, pages 167–174. ACM, 2007.

Satoshi Sekine and Chikashi Nobata. Definition, Dictionaries and Tagger for Extended Named Entity Hierarchy. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC '04)*, pages 1977–1980. ELRA, 2004.

Christian Sengstock and Michael Gertz. CONQUER: A System for Efficient Context-aware Query Suggestions. In *Proceedings of the 20th International Conference on World Wide Web (WWW '11)*, pages 265–268. ACM, 2011.

Andrea Setzer and Robert J. Gaizauskas. Annotating Events and Temporal Information in Newswire Texts. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC '00)*, pages 1287–1294. ELRA, 2000.

Milad Shokouhi. Detecting Seasonal Queries by Time-series Analysis. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*, pages 1171–1172. ACM, 2011.

Milad Shokouhi and Kira Radinsky. Time-sensitive Query Auto-completion. In *Proceedings of the 35th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '12)*, pages 601–610. ACM, 2012.

Mário J. Silva, Bruno Martins, Marcirio Silveira Chaves, Ana Paula Afonso, and Nuno Cardoso. Adding Geographic Scopes to Web Resources. *Computers, Environment and Urban Systems*, 30(4):378–399, 2006.

Sameer Singh, Amarnag Subramanya, Fernando Pereira, and Andrew McCallum. Large-scale Cross-document Coreference Using Distributed Inference and Hierarchical Models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL '11)*, pages 793–803. ACL, 2011.

Carlota S. Smith. The Syntax and Interpretation of Temporal Expressions in English. *Linguistics and Philosophy*, 2(1):43–99, 1978.

Catherine Soanes and Angus Stevenson, editors. *Oxford Dictionary of English*. Oxford University Press, Oxford, UK, 2nd edition, 2003.

Kathrin Spreyer and Anette Frank. Projection-based Acquisition of a Temporal Labeller. In *Proceedings of the 3rd International Joint Conference on Natural Language Processing (IJCNLP '08)*, pages 489–496. ACL, 2008.

Ralf Steinberger, Bruno Pouliquen, and Johan Hagman. Cross-Lingual Document Similarity Calculation Using the Multilingual Thesaurus EUROVOC. In *Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLing '02)*, pages 415–424. Springer, 2002.

Ralf Stoecker. Action Explanation. In Ernie Lepore and Kirk Ludwig, editors, *A Companion to Donald Davidson*, part I, chapter 1, pages 15–31. Wiley-Blackwell, New York, NY, USA, 2013.

Jannik Strötgen and Michael Gertz. HeidelTime: High Quality Rule-Based Extraction and Normalization of Temporal Expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval '10)*, pages 321–324. ACL, 2010a.

Jannik Strötgen and Michael Gertz. TimeTrails: A System for Exploring Spatio-Temporal Information in Documents. In *Proceedings of the 36th International Conference on Very Large Data Bases (VLDB '10)*, pages 1569–1572. VLDB Endowment, 2010b.

Jannik Strötgen and Michael Gertz. WikiWarsDE: A German Corpus of Narratives Annotated with Temporal Expressions. In *Proceedings of the Conference of the German Society for Computational Linguistics and Language Technology (GSCL '11)*, pages 129–134, 2011.

Jannik Strötgen and Michael Gertz. Event-centric Search and Exploration in Document Collections. In *Proceedings of the 12th ACM/IEEE Joint Conference on Digital Libraries (JCDL '12)*, pages 223–232. ACM, 2012a.

Jannik Strötgen and Michael Gertz. Temporal Tagging on Different Domains: Challenges, Strategies, and Gold Standards. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC '12)*, pages 3746–3753. ELRA, 2012b.

Jannik Strötgen and Michael Gertz. Multilingual and Cross-domain Temporal Tagging. *Language Resources and Evaluation*, 47(2):269–298, 2013a.

Jannik Strötgen and Michael Gertz. Proximity^2-aware Ranking for Textual, Temporal, and Geographic Queries. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM '13)*, pages 739–744. ACM, 2013b.

Jannik Strötgen, Michael Gertz, and Pavel Popov. Extraction and Exploration of Spatio-Temporal Information in Documents. In *Proceedings of the 6th Workshop on Geographic Information Retrieval (GIR '10)*, pages 16:1–16:8. ACM, 2010.

Jannik Strötgen, Michael Gertz, and Conny Junghans. An Event-centric Model for Multilingual Document Similarity. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*, pages 953–962. ACM, 2011.

Jannik Strötgen, Omar Alonso, and Michael Gertz. Retro: Time-Based Exploration of Product Reviews. In *Proceedings of the 34th European Conference on Advances in Information Retrieval (ECIR '12)*, pages 581–582. Springer, 2012a.

Jannik Strötgen, Omar Alonso, and Michael Gertz. Identification of Top Relevant Temporal Expressions in Documents. In *Proceedings of the 2nd Temporal Web Analytics Workshop (TempWeb '12)*, pages 33–40. ACM, 2012b.

Jannik Strötgen, Julian Zell, and Michael Gertz. HeidelTime: Tuning English and Developing Spanish Resources for TempEval-3. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval '13)*, pages 15–19. ACL, 2013.

Jannik Strötgen, Ayser Armiti, Tran Van Canh, Julian Zell, and Michael Gertz. Time for More Languages: Temporal Tagging of Arabic, Italian, Spanish, and Vietnamese. *ACM Transactions on Asian Language Information Processing (TALIP)*, 13(1):1:1–1:21, 2014a.

Jannik Strötgen, Thomas Bögel, Julian Zell, Ayser Armiti, Tran Van Canh, and Michael Gertz. Extending HeidelTime for Temporal Expressions Referring to Historic Dates. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC '14)*, pages 2390–2397. ELRA, 2014b.

Bernhard Suhm, Lori Levin, Noah Coccaro, Jamie Carbonell, Ryosuke Isotani, Alon Lavie, Laura Mayfield, Carolyn P. Rosé, Carol Van Ess-Dykema, and Alex Waibel. Speech-language Integration in a Multilingual Speech Translation System. In *Proceedings of the Workshop on Integration of Natural Language and Speech Processing*. AAAI, 1994.

Xavier Tannier. Extracting News Web Page Creation Time with DCTFinder. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC '14)*, pages 2037–2042, 2014.

Tao Tao and ChengXiang Zhai. An Exploration of Proximity Measures in Information Retrieval. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*, pages 295–302. ACM, 2007.

Benjamin E. Teitler, Michael D. Lieberman, Daniele Panozzo, Jagan Sankaranarayanan, Hanan Samet, and Jon Sperling. NewsStand: A New View on News. In *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS '08)*, pages 144–153. ACM, 2008.

Carol L. Tenny and James Pustejovsky. A History of Events in Linguistic Theory. In Carol L. Tenny and James Pustejovsky, editors, *Events as Grammatical Objects: The Converging Perspectives of Lexical Semantics and Syntax*. CSLI, Stanford, CA, USA, 2000a.

Carol L. Tenny and James Pustejovsky, editors. *Events as Grammatical Objects: The Converging Perspectives of Lexical Semantics and Syntax*. CSLI, Stanford, CA, USA, 2000b.

Laurence C. Thompson. *A Vietnamese Reference Grammar*. University of Hawaii Press, Honolulu, HI, USA, 1991.

Frank Tobian. Modell und Rankingverfahren zur Kombination von textueller und geographischer Suche. Bachelor's thesis, Heidelberg Universtiy, 2011.

Katrin Tomanek, Joachim Wermter, and Udo Hahn. Sentence and Token Splitting Based on Conditional Random Fields. In *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics (PACLING '07)*, pages 49–57. PACL, 2007.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. Feature-rich Part-of-speech Tagging with a Cyclic Dependency Network. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL '03)*, pages 173–180. ACL, 2003.

Edward R. Tufte. *Beautiful Evidence*. Graphics Press, Cheshire, CT, USA, 2006.

E. Lynn Usery. A Feature-based Geographic Information System Model. *Photogrammetric Engineering and Remote Sensing*, 62(7):833–838, 1996.

Naushad UzZaman and James F. Allen. TRIPS and TRIOS System for TempEval-2: Extracting Temporal Information from Text. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval '10)*, pages 276–283. ACL, 2010.

Naushad UzZaman and James F. Allen. Event and Temporal Expression Extraction From Raw Text: First Step Towards a Temporally Aware System. *International Journal of Semantic Computing*, 4(4):487–508, 2011.

Naushad UzZaman, Hector Llorens, Leon Derczynski, James F. Allen, Marc Verhagen, and James Pustejovsky. SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval '13)*, pages 1–9. ACL, 2013.

Matje van de Camp and Henning Christiansen. Resolving Relative Time Expressions in Dutch Text with Constraint Handling Rules. In *Proceedings of the 7th International Workshop on Constraint Solving and Language Processing (CSLP '12)*, pages 74–85. Springer, 2012.

Marc Verhagen. Tempeval2 Data – Release Notes. Brandeis University, 2011. URL `http://timeml.org/site/timebank/tempeval/tempeval2-data.zip` [last accessed October 10, 2014].

Marc Verhagen and Jessica Moszkowicz. AQUAINT TimeML 1.0 Corpus Documentation, 2008. ULR `http://www.timeml.org/site/timebank/aquaint-timeml/aquaint_timeml_1.0.tar.gz` [last accessed October 10, 2014].

Marc Verhagen and James Pustejovsky. Temporal Processing with the TARSQI Toolkit. In *Proceedings of the 22nd International Conference on on Computational Linguistics (COLING '08)*, pages 189–192. ACL, 2008.

Marc Verhagen and James Pustejovsky. The TARSQI Toolkit. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC '12)*, pages 2043–2048. ELRA, 2012.

Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Graham Katz, and James Pustejovsky. SemEval-2007 Task 15: TempEval Temporal Relation Identification. In *Proceedings of the 4th International Workshop on Semantic Evaluation (SemEval '07)*, pages 75–80. ACL, 2007.

Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Jessica Moszkowicz, and James Pustejovsky. The TempEval Challenge: Identifying Temporal Relations in Text. *Language Resources and Evaluation*, 43(2):161–179, 2009.

Marc Verhagen, Roser Saurí, Tommaso Caselli, and James Pustejovsky. SemEval-2010 Task 13: TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval '10)*, pages 57–62. ACL, 2010.

María Teresa Vicente-Diez and Paloma Martinez. Temporal Semantics Extraction for Improving Web Search. In *Proceedings of the 20th International Workshop on Database and Expert Systems Application (DEXA '09)*, pages 69–73. IEEE, 2009.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. ACE 2005 Multilingual Training Corpus. Linguistic Data Consortium (LDC), Philadelphia, PA, USA, 2006. URL `https://catalog.ldc.upenn.edu/LDC2006T06` [last accessed October 10, 2014].

Chuang Wang, Xing Xie, Lee Wang, Yansheng Lu, and Wei-Ying Ma. Detecting Geographic Locations from Web Resources. In *Proceedings of the 2nd Workshop on Geographic Information Retrieval (GIR '05)*, pages 17–24. ACM, 2005a.

Chuang Wang, Xing Xie, Lee Wang, Yansheng Lu, and Wei-Ying Ma. Web Resource Geographic Location Classification and Detection. In *Proceedings of the 14th International Conference on World Wide Web (WWW '05)*, pages 1138–1139. ACM, 2005b.

Rui Wang and Günter Neumann. Ontology-Based Query Construction for GeoCLEF. In *Proceedings of the 9th Workshop of the Cross-Language Evaluation Forum (CLEF '08)*, pages 880–884. Springer, 2009.

Yafang Wang, Bin Yang, Spyros Zoupanos, Marc Spaniol, and Gerhard Weikum. Scalable Spatio-temporal Knowledge Harvesting. In *Proceedings of the 20th International Conference on World Wide Web (WWW '11)*, pages 143–144. ACM, 2011.

Sholom M. Weiss, Nitin Indurkhya, Tong Zhang, and Fred Damerau. *Text Mining: Predictive Methods for Analyzing Unstructured Information*. Springer, New York, NY, USA, 2005.

Wolodja Wentland, Johannes Knopp, Carina Silberer, and Matthias Hartung. Building a Multilingual Lexical Resource for Named Entity Disambiguation, Translation and Transliteration. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC '08)*, pages 3230–3237. ELRA, 2008.

Allison Gyle Woodruff and Christian Plaunt. GIPSY: Automated Geographic Indexing of Text Documents. *Journal of the American Society for Information Science*, 45(9):645–655, 1994.

Mitsuo Yamamoto, Yuku Takahashi, Hirotoshi Iwasaki, Satoshi Oyama, Hiroaki Ohshima, and Katsumi Tanaka. Extraction and Geographical Navigation of Important Historical Events in the Web. In *Proceedings of the 10th International Symposium on Web and Wireless Geographical Information Systems (W2GIS '11)*, pages 21–35. Springer, 2011.

Jie Yin, Derek Hao Hu, and Qiang Yang. Spatio-temporal Event Detection Using Dynamic Conditional Random Fields. In *Proceedings of the 21st International Jont Conference on Artifical Intelligence (IJCAI '09)*, pages 1321–1326. Morgan Kaufmann, 2009.

Bo Yu and Guoray Cai. A Query-aware Document Ranking Method for Geographic Information Retrieval. In *Proceedings of the 4th Workshop on Geographic Information Retrieval (GIR '07)*, pages 49–54. ACM, 2007.

Yisong Yue, Thomas Finley, Filip Radlinski, and Thorsten Joachims. A Support Vector Method for Optimizing Average Precision. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*, pages 271–278. ACM, 2007.

Vanni Zavarella and Hristo Tanev. FSS-TimEx for TempEval-3: Extracting Temporal Information from Text. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval '13)*, pages 58–63. ACL, 2013.

Kuo Zhang, Juan Zi, and Li Gang Wu. New Event Detection Based on Indexing-tree and Named Entity. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*, pages 215–222. ACM, 2007.

Wei Vivian Zhang, Benjamin Rey, Eugene Stipp, and Rosie Jones. Geomodification in Query Rewriting. In *Proceedings of the 3rd Workshop on Geographic Information Retrieval (GIR '06)*, pages 23–27. ACM, 2006.

Ran Zhao, Quang Do, and Dan Roth. A Robust Shallow Temporal Reasoning System. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT '12)*, pages 29–32. ACL, 2012.

Yinghua Zhou, Xing Xie, Chuang Wang, Yuchang Gong, and Wei-Ying Ma. Hybrid Index Structures for Location-based Web Search. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management (CIKM '05)*, pages 155–162. ACM, 2005.

Slavoj Žižek. *Event: Philosophy in Transit.* Penguin, London, UK, 2014.