HITS' Graph-based System at the NTCIR-9 Cross-lingual Link Discovery Task

Angela Fahrni Vivi Nastase Michael Strube

Heidelberg Institute for Theoretical Studies
Schloss-Wolfsbrunnenweg 35
69118 Heidelberg, Germany
angela.fahrni@h-its.org

ABSTRACT

This paper presents HITS' system for the NTCIR-9 cross-lingual link discovery task. We solve the task in three stages: (1) anchor identification and ambiguity reduction, (2) graph-based disambiguation combining different relatedness measures as edge weights for a maximum edge weighted clique algorithm, and (3) supervised relevance ranking. In the file-to-file evaluation with Wikipedia ground-truth the HITS system is the top-performer across all measures and subtasks (English-2-Chinese, English-2-Japanese and English-2-Korean). In the file-2-file and anchor-2-file evaluation with manual assessment, the system outperforms all other systems on the English-2-Japanese subtask and is one of the top-three performing systems for the two other subtasks.

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*text analysis*; I.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*linguistic processing*

General Terms

Experimentation.

Keywords

Wikipedia, Cross-lingual Link Discovery, Anchor Identification, Link recommendation. **Team Name:** [HITS]

Subtasks/Languages: [Japanese Cross-lingual Link Discovery Task][Korean Cross-lingual Link Discovery Task][Chinese Cross-lingual Link Discovery Task]
External Resources Used: [Chinese Wikipedia] [Japanese Wikipedia] [English Wikipedia] [German Wikipedia] [Italian Wikipedia] [French Wikipedia] [Russian Wikipedia] [Dutch Wikipedia] [TreeTagger]

1. INTRODUCTION

Link discovery is the task of automatically inserting links between documents. In recent years, several approaches for link insertion relative to Wikipedia – also known as *wikification* or *wikifying* – have been proposed ([3, 10]). While current systems operate in a monolingual scenario, cross-lingual link discovery extends this task by adding a further challenge: instead of linking text anchors to Wikipedia articles of the same language, they are linked to Wikipedia articles in other languages. This may enable Wikipedia users who do not understand the language well to find information in their own language and may enhance knowledge access across language boundaries.

Cross-lingual link discovery comprises three main challenges. *Anchor identification* is the problem of deciding if a text string serves as an appropriate anchor and if so where the boundaries of the anchor are. The following examples illustrates these two phenomena:

(1) *Most* of the immigrants are skilled.

(2) Some schools believe that competition [...] gives a sense of good *sportsmanship*.

(3) The *western style* is seen in a long stirrup length [...], an upright posture [...] and the distinctive one-handed hold on the reins [...].

In example (1) the token *Most* is not supposed to be linked, as it expresses here a quantity and does not refer to the Czech city *Most*. While in example (2) the adjective *good* does not belong to the anchor *sportsmanship*, the adjective *western* in example (3) is part of the anchor *western style*.

As many anchors are highly ambiguous and can be used to refer to different Wikipedia articles, *disambiguation in context* is inevitable. The anchor *culture* may refer among others to the Wikipedia page *cell culture*, the article on *culture* or to the article about *organizational culture*.

Finally, only the most related Wikipedia pages for a given topic should be linked. Performing a *relevance ranking* of the identified Wikipedia article/anchors mappings allows to select the most important ones. In a text on *martial arts* a link to the Wikipedia page *country of origin* is not as informative as a link to *kickboxing*.

Our system implements anchor identification, disambiguation and relevance ranking. We focus on disambiguation and propose a maximum weighted clique algorithm integrating different relatedness measures as edge weights. To link from English to other languages, i.e. Chinese, Japanese and Korean, we exploit cross-language links in Wikipedia and enhance them using image and hyperlink information.

The remainder of the paper is organized as follows: In Section 2 we discuss related work. Our approach is presented in Section 3 and the experiments are analyzed in Section 4.

2. RELATED WORK

Existing approaches to annotate texts with links to Wikipedia do not just differ in their methods, but also regarding their aims. The first group of systems links only a few keywords in a text to Wikipedia.

The *Wikify!* system [3] approaches the keyword linking

task in two stages. The first step identifies keywords/keyphrases using *key-phraseness* – a metric that captures the probability of a term being linked in a Wikipedia page. The identified keywords are then disambiguated relative to Wikipedia pages. Two disambiguation methods were tested: (i) a supervised approach that uses existing hyperlinks in Wikipedia for training and represents instances using local features (surrounding words and their part of speech tags); (ii) an unsupervised approach that uses a Lesk relatedness measure, that relies on the overlap of context with the potential Wikipedia article target. The supervised method performs best on a random selection of Wikipedia articles, considering the existing hyperlinks as the gold standard.

Wikipedia Miner [10] implements a supervised disambiguation approach using hyperlinks in articles as training data. Each instance is represented through several features: the prior probability of a term to be linked to a certain concept (computed over the entire Wikipedia); relatedness measures based on the link structure of Wikipedia, that capture average relatedness of a potential Wikipedia article to the words in its context. This is a way of integrating the larger context into the model. We combine this proposed relatedness measures with others and use them to weigh the edges in our graph. Some work (e.g. [2, 4], TAC's entity linking task[1]) focuses on the linkage of named entities such as persons, places and organizations to Wikipedia. The entity disambiguation system proposed by [4] employs a vector space model. To disambiguate named entities relative to Wikipedia, the similarity values between vector representations of the respective Wikipedia pages and the vector representation derived from the document to be analyzed are compared. The vectors contain contextual word level and Wikipedia category information. This work is closely related to our term recognition and filtering component which incorporates similarities of vector representations as features for a supervised model.

The third group of work (e.g. [7, 19, 5, 22, 15]) aims to link as many strings in a text as possible to Wikipedia articles and not just a few selected keywords or named entities. This leads to a semantic representation of text. In this case, Wikipedia is used as a sense inventory (instead of, e.g., WordNet).

A global disambiguation method is proposed by [7]: to determine the best combination of entities (Wikipedia articles) for a text a score which combines local context-entity compatibility (node potential) and global coherence between entities (clique potential) is maximized. This global disambiguation approach improves over [10], which performs term-by-term disambiguation. We also employ a global approach and consider more information to model global coherence.

In [15], a two pass approach is presented. First, a supervised model using local contextual features predicts a Wikipedia article for each text string. In a second pass, these predictions are used to calculate relatedness in a similar way as [10]. In contrast to our system, where a supervised filtering step supports the anchor term identification and narrows down the search space for the graph-based approach, [15] employ the predictions of the first pass as fixed points to calculate relatedness.

[19] describe an expanded Hidden Markov Model to model interleaved chains for disambiguation relative to Wikipedia. To determine the chains they use a semantic relatedness measure, and find the best sequence of concepts relative to the text. *TagMe* [5] is tuned to disambiguate short text passages – snippets of search engine queries, tweets, etc. [22] use in addition to Wikipedia other knowledge sources like query logs to extract features.

While all these previous approaches work on a monolingual level, mainly on English, the NTCIR-9 cross-lingual link discovery shared task captures a cross-lingual perspective.

[[In]$_{31}$ [[popular]$_{24}$ [culture]$_{186}$]$_1$]$_{12}$, [[the]$_{12}$ [term]$_{41}$]$_3$ "[[Martial]$_{13}$ [Arts]$_{22}$]$_6$"

In [popular culture]$_1$, the term "[Martial Arts]$_2$"

Figure 1: Recognized ambiguous anchors before (top) and after (bottom) anchor identification and ambiguity reduction: Brackets mark anchor boundaries, indices show the number of ambiguities.

3. METHODOLOGY

In this section, we first describe the system architecture, then we explain our knowledge base extracted from Wikipedia and the different components in greater detail.

3.1 System Architecture

The HITS system consists of four components, as illustrated in Figure 2. (1) An input text in the YAWN XML format is preprocessed [16]. The XML is parsed in order to receive clean text. Structural information about paragraphs, tables and formatting is kept. To store all this information we use the multi-layer annotation objects specified by the EU project CoSyne[2]. The clean text parts are then tokenized.

(2) Anchor terms are identified and for each of them all corresponding English Wikipedia articles are retrieved from our lexicon. As the top of Figure 1 illustrates, anchors can overlap each other and are sometimes highly ambiguous. To decide on term boundaries and to exclude Wikipedia pages which are not likely given the context of an anchor, we employ some heuristics and a supervised model. After the filtering step, the boundaries of the anchors are determined and there are less candidate Wikipedia pages for each remaining anchor as shown at the bottom of Figure 1.

(3) We perform disambiguation using a graph-based approach. In this step, all ambiguities are resolved and for each anchor exactly one English Wikipedia article is selected. Until this point, we work on a monolingual level. As the aim is to insert links to Japanese, Korean or Chinese Wikipedia articles, the corresponding articles for the required language have to be identified. We retrieve them from our knowledge base derived from Wikipedia (see Section 3.2). Note, we disambiguate all anchors, even if we cannot find an equivalent Wikipedia article in the target language (e.g. Japanese) for the respective English Wikipedia article in our knowledge base. As we pursue a disambiguation approach where the anchors disambiguate each other, disambiguating anchors that will not appear in the end results helps to disambiguate anchors which have a corresponding Wikipedia article in the target language.

(4) As for each document at most 250 anchors are allowed, we rank the remaining anchors with a corresponding Wikipedia article in the target language according to relevance.

[1] http://nlp.cs.qc.cuny.edu/kbp/2011/

[2] http://sourceforge.net/projects/cosyne-eu/

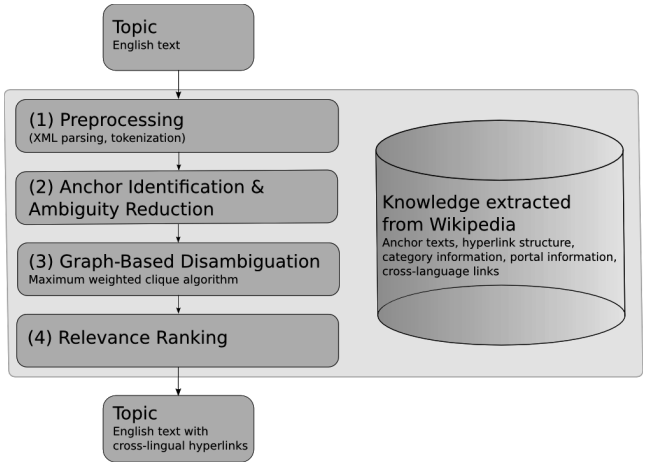

Figure 2: System architecture: Knowledge extracted from Wikipedia is used in steps (2) to (4)

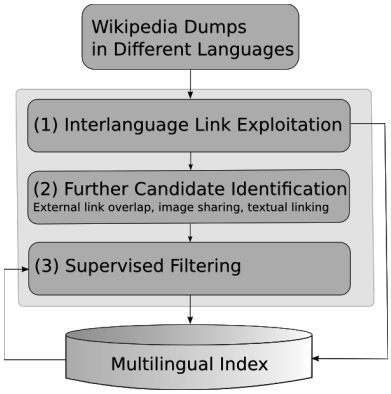

Figure 3: Building the multilingual index

3.2 Multilingual Concept Repository

Following previous work (e.g. [8, 11]), we understand each Wikipedia article as corresponding to a concept while the content of the article is seen as the description of the concept. Each concept has several English lexicalizations, which we derive from anchors, redirects and disambiguation pages in Wikipedia. Our knowledge base contains among others the following English lexicalizations for the concept *Chinese martial arts*: *martial arts*, *Chinese martial arts*, *martial artists*, *kungfu*, *Gong Fu*, *Chinese boxing*. Thanks to Wikipedia's multilinguality we are able to enhance this monolingual concept repository with concept realizations and descriptions from several languages and build a multilingual resource. We create a multilingual index which contains for each concept the names of the corresponding Wikipedia articles in various languages and a confidence score for the mapping. In Table 1, the identified corresponding Korean, Chinese and Japanese Wikipedia article names and confidence scores for the concepts *Chinese martial arts*, *Diphyllobothrium* and *La Fayette class frigate* are listed. As the Korean example implies, it is a many to many mapping.

3.2.1 Building a Multilingual Index

To build the multilingual index, we proceed as depicted in Figure 3. First, we extract all cross-language links pointing from the English Wikipedia to one of the target languages and vice versa. The output of this step is a list of candidate mappings between English and the three target languages Chinese, Japanese and Korean. If there is a one-to-one mapping between a page in English and one in a target language we directly add this pair to the multilingual index assigning it a confidence score of 1.0. All others, i. e. one-to-many or many-to-many mappings, are appended to a list of candidate mappings. To enhance coverage, we additionally process several language versions of Wikipedia[3] and apply a triangulaton method (similar to [20]): given three Wikipedia pages A, B and C in three different languages and a cross-language link pointing from A to B and one from B to C, we also establish a link between A and C. As for the direct cross-lingual links we add one-to-one mappings to the multilingual index with confidence value 1.0, one-to-many and many-to-many ones to the candidate list. While for example the English Wikipedia article *La Fayette class frigate* and the Japanese and Korean versions are linked by a direct cross-language link, the link between the English and the Chinese page is mediated by the respective Russian article. To retrieve more candidate pairs, we also process other information sources which may indicate a mapping between two pages even if there exists no cross-language links:

1. **External hyperlinks**: If pages from different language versions share some external links, they are likely to deal with the same thing.
2. **Images**: Pages containing the same image tend to be similiar as well. The English article *Diphyllobothrium* and the Japanese version for example share the same image (see Table 1).
3. **Templates**: Sometimes the English name or word is mentioned in the Chinese, Korean or Japanese Wikipedia article and the other way around using a template such as the *Nihongo template*. If we can uniquely map the name or word in the foreign language to a Wikipedia page, we also count the respective pair of Wikipedia pages as candidate pair.

In order to reduce the noise in the list of candidate mappings, a supervised filtering technique is applied: for each target language a binary classifier is trained on instances derived from the multilingual index and by using features such as a relatedness measure based on link overlap. For each English article the highest ranked mappings according to the confidence value returned by the classifier are added to the multilingual index.

Table 2 shows a quantitative evaluation of this mapping process: the first two rows indicate the coverage by using direct interlanguage links pointing from the target language to the English Wikipedia and vice versa: while *Fract EN CLD* exhibits the fraction of English Wikipedia articles with a mapping to the target language, *Fract L CLD* presents the fraction of Wikipedia articles in the target language with a mapping to the English Wikipedia. The last two rows (*Fract EN Total, Fract L Total*) report the coverage after applying the described mapping procedure. Compared to just using direct interlanguage links, we gain mappings for more than

[3]We process the following language versions: English (2011/01/15), Chinese (2011/06/23), Japanese (2010/11/02), Korean (2011/06/21), German (2011/01/11), Italian (2011/01/30), French (2011/02/01), Russian (2011/07/16), Dutch (2011/01/26).

Table 1: English, Japanese, Chinese and Korean Wikipedia article names with confidence scores

EN	JA	ZH	KO
Chinese martial arts	中国武術:1.0 (Kung fu)	中国武术:1.0 (Kung fu)	쿵후:0.9989 (Kung fu) 무예:0.9989 (Martial arts)
Diphyllobothrium	広節裂頭条虫:0.99 (Diphyllobothrium latum)		
La Fayette class frigate	ラファイエット級フリゲート:1.0	康定級巡防艦:1.0	라파예트급 프리깃함:1.0

Table 2: Coverage of the multilingual index

	EN-to-JA	EN-to-KO	EN-to-ZH
Fract EN CLD	0.08	0.03	0.05
Fract L CLD	0.41	0.63	0.52
Fract EN Total	0.12	0.04	0.08
Fract L Total	0.47	0.69	0.59

40,000 (KO) / 90,000 (ZH) / 100,000 (JA) English Wikipedia articles. We did not systematically evaluate the quality of the mappings. To illustrate the mapping process, e.g., there is no direct cross-language link between the English page on *Chinese martial arts* and a Korean page. Using the described approach, the top ranked Korean pages for this English page are

- *쿵후 (Kung Fu)*, and
- *무예 (Martial arts)* followed by
- *우슈 (Wushu)* (0.987),
- the movie *쿵푸 허슬 (Kung Fu Hustle)* (0.982), and
- *격투 스포츠 (Fighting sports)* (0.614).

We exclude all test topics from the mapping process.

3.2.2 Knowledge Extracted from Wikipedia

The ambiguity reduction and disambiguation steps are both highly informed by knowledge derived from Wikipedia. We extract the following information from the English Wikipedia:

Incoming and outgoing links: We do not include incoming links from list articles as these collectional articles are conceptually different from others.

Incoming links from list articles: For each concept all list articles in which it appears are identified.

Categorial information: For each concept we extract all categories which do not have an administrative purpose from the English Wikipedia and also include categories that are at most three steps above in the category hierarchy.

Portal information: In Wikipedia, articles on the same subject are often summarized under a portal. There is, e.g., a portal on *martial arts*. Portals point to all relevant categories for the given subject. We extract these categories and expand them to retrieve the subsumed articles.

None of the test topics are considered here.

3.3 Anchor Recognition and Supervised Ambiguity Reduction

The first step in the process is the anchor recognition and the identification of candidate concepts (see Figure 2). Whenever we find an n-gram in the cleaned text that is in our lexicon, we check if the following condition is met: the keyphraseness of the n-gram, i.e. the ratio between how many times an n-gram is linked and how many times it appears in the English Wikipedia [3], must exceed a certain threshold[4]. This restriction prevents linking tokens such as *a* or *be*. If this constraint is satisfied, all possible candidate concepts with a prior probability higher than a threshold[5] for that n-gram are retrieved from the lexicon. The prior probability for a concept c given an anchor a is defined as

$$p(c|a) = \frac{count(a_c)}{\sum_{a_i \in C_a} count(a_i)} \quad (1)$$

where $count(a_c)$ is the number of times anchor a is linked to concept c in Wikipedia and C_a the set of candidate concepts for anchor a.

To decide on the anchor boundaries (see Figure 1) and to reduce the ambiguity, a binary classifier is trained with the two classes *correct concept* and *wrong concept* for an anchor given its context. For training, we extracted all links to other Wikipedia articles from 300 featured Wikipedia articles we randomly selected[6]. The positive instances are derived from these extracted links, while the negative examples are deduced by randomly selecting other candidate Wikipedia articles from our lexicon for the anchors of these links. The used features describe three aspects:

1. The **prior probability** (Equation 1) expresses the likelihood of a concept given a certain anchor. The anchor *staff* for example refers more often to *Employee* ($p = 0.22$) than to the concept *Gun (staff)* ($p = 0.02$).
2. **Prominence of a concept** is approximated by the following two features:
   a. *Concept prior probability* defined by the ratio between the number of Wikipedia pages that link to a certain concept and the total number of Wikipedia articles.
   b. The *Hits probability* is the fraction of times an article has been viewed by a user in the past six months.[7]

   The assumption is that anchors tend to refer to more prominent concepts. It is for example a priori more likely that an anchor refers to the more prominent concept *Wushu (sport)* than to *Wushu (role-playing game)*.

[4]We empirically set the threshold to $t = 0.01$.
[5]The threshold is empirically set to $p = 0.01$.
[6]Wikipedia articles tagged as *featured* are supposed to be of high quality.
[7]We derived these numbers by downloading the hits count for January until June 2011 from `http://dammit.lt/wikistats/`.

3. **Context fit**: In a text about chinese martial arts, it becomes for example more probable that the anchor *staff* refers to *Gun (staff)* instead of *Employee*. Similiar to [10] all unambiguous anchors serve as context. Although this approach is problematic as it is not guaranteed that unambiguous anchors are present [15], we assume that our texts are long enough and contain a couple of unambiguous anchors. Given these context concepts we build four context vectors:

   a. *Category context vector:* it contains the weights for each category retrieved for the context concepts from our category index. The weights for each category are determined by the fraction of context concepts that are associated with this category. For the other context vectors the weights are calculated analogously.

   b. *Concept context vector:* it consists of the weights for all context concepts as well as their incoming and outgoing links

   c. *List context vector:* it holds the weights of the context concept's incoming links from lists.

   d. *Portal context vector:* it comprises the weights for the portals associated with the context concepts.

   For each candidate concept for a certain anchor, a category, a concept, a list and a portal vector is built. The weights are 1 or 0 depending on the presence or absence of the respective category, concept, list or portal in the corresponding context vector. As context features we use the cosine similarities between the context vectors and the vectors of the candidate concept. In addition, the category, concept, list and portal of the candidate concept that has the highest and the lowest weight in the respective context vector are identified. The respective weights are taken as additional features.

We apply Weka's decision tree classifier J48 [21]. Instead of the class value returned by the classifier, the class membership probability is used. All candidate concepts with a class membership probability higher than a threshold[8] are kept. If the remaining anchors overlap each other, the longest one is chosen and in case of ties the one standing most right as this is most likely the head in an English noun phrase.

3.4 Graph-based Disambiguation: Maximum Edge Weighted Clique Algorithm

We implement a two step method for the joint selection of a concept for each of the remaining anchors in a text. The first step is supervised (Section 3.5) in which we learn to predict the strength of the connection between two concepts. The prediction relies on several relatedness measures between concepts. The second step is a graph-based disambiguation method, where the edges in the graph are weighted by the score predicted in step 1. Determining the best combination of concepts is done by identifying the *maximum edge weighted clique* (Section 3.6).

3.5 Learning the Edge Weights

In previous work on Wikipedia various relatedness and similarity measures have been proposed [13, 9, 6], each different, yet each showing good results on classical test sets and NLP tasks – coreference resolution [13], text categorization [6], concept identification [10]. This shows that the rich structure and information in Wikipedia provides different perspectives of the relatedness between concepts. Because of this, our approach combines different relatedness scores into one that describes the strength of the connection between two concepts. We do this by learning a model from concept co-occurrences. Each instance represents a pair of concepts through several features that capture the strength of the relation between the pair from different perspectives:

A relatedness measure based on *incoming links* [10]. Incoming links for a concept c_A are hyperlinks that "point to" the page corresponding to c_A. This measure captures first-order co-occurrence information at the concept-level – the more pages link to both c_A and c_B, the higher the value:

$$rel_{in}(c_A, c_B) = \frac{log(max(|A|, |B|)) - log(A \bigcap B)}{log(|W|) - log(min(|A|, |B|))}$$

A and B are the sets of c_A's and c_B's incoming links respectively, and W is the set of Wikipedia concepts.

A relatedness measure based on *outgoing links* [10]. Outgoing links for a concept c_A are hyperlinks that originate on the page corresponding to c_A. This measure captures a simplified version of second order co-occurrence information – it relies on the extent to which concepts that appear in c_A's page also occur in c_B's page:

$$rel_{out}(c_A, c_B) = \cos(OutW_A \cdot OutW_B)$$

$OutW_A$ and $OutW_B$ are weighted vectors of outgoing links for c_A and c_B respectively. A weight is the logarithm of the inverse frequency of the respective outgoing link: the more often a concept is linked in Wikipedia, the less discriminative it is and the smaller its weight.

A relatedness measure based on *categorial information.* Categories are assigned by Wikipedia contributors, and group pages that have something in common. Hence, pages under the same category are related. We compute this as the cosine similarity between the vectors of the extended parent categories of concepts c_A and c_B:

$$rel_{cat}(c_A, c_B) = \cos(CW_A \cdot CW_B)$$

where CW_A and CW_B are two vectors containing the weights of c_A's and c_B's extended parent categories, respectively. A weight is the logarithm of the inverse frequency of the respective category. The assumption is that the less frequent a parent category is, the more informative it is if both concepts c_A and c_B are associated with it.

The preference of a concept for a context anchor's disambiguation. For two anchors to be disambiguated, t_A and t_B, we compute how much the disambiguation c_A for term t_A prefers the disambiguation c_B for anchor t_B:

$$pref_{AB}(c_A, c_B|t_B) = \frac{count(c_A, c_B)}{\sum_{c_j \in C_{t_B}} count(c_A, c_j)}$$

C_{t_B} is the set of concepts that anchor t_B may refer to, and $c_j \in C_{t_B}$. $count(c_A, c_j)$ is the number of times the concept pair (c_A, c_j) occurs. $pref_{BA}(c_B, c_A|t_A)$ is a feature as well.

The purpose of this learning step is to predict the strength of the association between two concepts. For the training data we compute an approximation of this measure as the

[8] We employed two different thresholds: $t = 0.4$ for HITS_..._01, $t = 0.1$ for HITS_..._02 and HITS_..._03.

co-occurrence probability of the two concepts, given their corresponding anchors in the text t_A and t_B:

$$coocP(c_A, c_B) = e^{p(c_A,c_B|t_A,t_B)-chance(t_A,t_B)-1}$$

$$p(c_A, c_B|t_A, t_B) = \frac{count(c_A, c_B)}{\sum_{c_i \in C_{t_A}, c_j \in C_{t_B}} count(c_i, c_j)}$$

$$chance(t_A, t_B) = \frac{1}{|C_{t_A}| \times |C_{t_B}|}$$

C_{t_A} and C_{t_B} have the same meaning as above. This measure takes into account the ambiguity of the terms to be disambiguated, and quantifies the strength of the association between one specific interpretation of the two concepts considering all other options. $p(c_A, c_B|t_A, t_B)$ quantifies the absolute strength of the c_A, c_B pair, and we deduct from this the $chance(t_A, t_B)$. The reason for this is that if all concept pairs are equally likely, it means that none are really informative, and as such should have low strength. -1 is deducted to map the function to the [0,1] interval.

As not each occurrence of an anchor in a Wikipedia page is marked as a hyperlink, this measure is only an approximation for the true co-occurrence probability of a concept pair given the respective terms.

These features represent each concept pair instance. The co-occurrence probability is what the model learns to predict. Because features and prediction are numeric, and we envisage the score to combine the values of the provided features, linear regression is a natural choice. Learning is done using Weka's Linear Regression [21]. Training instances are generated in the same way as for the filtering step, but for pairs of links instead of single links.

3.6 Concept Disambiguation as a Maximum Edge Weighted Clique Problem

We represent a text as a complete n-partite graph $G = (V_1, ..V_n, E)$. Each partition V_i corresponds to an anchor t_i in the text, and contains as vertices all concepts c_{i_j} for anchor t_i that remained after the first filtering step. Each vertex from a partition is connected to all vertices from the other partitions (making the graph complete) through edges $e_{v_i,v_j} \in E$ whose weights w_{v_i,v_j} are determined by the model learned in the previous step. In this graph we want to determine the maximum edge weighted clique.

A clique is a subgraph in which each vertex is connected to all other vertices [12]. A maximum clique of a graph is the clique with the highest cardinality. Given our n-partite graph G a maximum clique contains for each partition (anchor t_i) exactly one vertex (concept). A maximum edge weighted clique is the clique C with the highest edge weights sum $W_e(C)$ [14]:

$$W_e(C) = \sum_{v_i,v_j \in C} w_{v_i,v_j}$$

Identifying the maximum weighted clique of a graph is an NP-complete problem, but several approximations have been proposed (see [14, 1] for an overview). We apply an adapted beam search algorithm to approximate the maximum edge weighted clique. For each anchor the concept which corresponds to the vertex which is part of the clique in this partition is selected. For each anchor/concept pair, the corresponding Wikipedia article in the target language is retrieved from the multilingual index. If there is no such article, the anchor is discarded from the solution.

3.7 Relevance Ranking

To select at most 250 anchors per text, we rank the remaining Wikipedia article/anchors pairs by relevance. As a Wikipedia article should only be linked once per document – the first time it is referred to –, all anchors that point to the same article are pooled together; subject to the ranking algorithm are therefore Wikipedia articles and their respective anchors. Note, we perform the ranking on the English side, but only for the English Wikipedia articles with a corresponding article in the target language. The approach is similiar to the process proposed by [10] to identify keywords.

We trained a binary classifier with the classes *is keyword* and *is no keyword*. Training instances are gathered from 100 *featured* Wikipedia articles disambiguated by the methods from Sections 3.3 and 3.4. The remaining anchors that are also hyperlinked in Wikipedia are positive instances, all others negative ones. The features describe six aspects:

1. **Key-phraseness** helps to select keywords [3, 10]. We take the average key-phraseness calculated over all anchors that refer to the Wikipedia article in question and the maximum key-phraseness [10].
2. The **prominence** of a Wikipedia article is approximated using the Wikipedia concept prior probability and the hits probability (see Section 3.3).
3. According to [10] **positional features** are helpful. The relative first and last occurrences as well as the relative distance between the two serve as further features.
4. **Structural information** is often telling: words – and the concepts they refer to – that are highlighted or part of a title are usually more important than others. We use the following frequencies as features: number of times a respective anchor appears in the whole text, in titles, in lists, in tables, in captions of images and in highlightings.
5. The **specifity** of a Wikipedia article also plays a role [10]. The more specific the more informative and worth to be linked to is an article. Here specifity is measured using the average, maximum and minimal information content calculated over all parent categories of an article as defined by [17]. The features describing specifity are only used in the runs with ID HITS_..._03.
6. **Relatedness**: The closer related a Wikipedia article is to the main subject the more likely it is that it is linked. We use the class membership probability and parts of the context features from the filtering step (see Section 3.3), namely cosine similarity between the portal and concept vectors and the respective maximum weights, as features.

As classifier's we use Weka's decision trees J48 [21]. We order the Wikipedia article/anchors pairs according to the class membership probability.

4. EXPERIMENTAL RESULTS

For each subtask, HITS submitted three runs. Table 4 describes the differences between the runs. Table 3 summarizes the results of the HITS system for various evaluation metrics [18]. The table includes the file-2-file evaluation with Wikipedia ground truth and manual assessment as well as the anchor-2-file evaluation with manual assessment. In addition to the results achieved by the HITS system, the table also contains the best scores for each evaluation metric. These best scores are independently selected for each evalu-

Table 3: Results of the HITS system compared to the best scores for various evaluation metrics. The best scores are independently selected for each evaluation metric, i. e. they are not produced by one single system. Best scores achieved by HITS are highlighted in bold.

File-2-File Evaluation with Wikipedia ground truth												
Run IDs	**Japanese**				**Korean**				**Chinese**			
	MAP	**r-prec**	**p5**	**p50**	**MAP**	**r-prec**	**p5**	**p50**	**MAP**	**r-prec**	**p5**	**p50**
Best Scores	0.316	0.409	0.840	0.629	0.447	0.513	0.848	0.521	0.373	0.471	0.832	0.581
01	0.310	0.403	0.816	0.626	**0.447**	0.509	**0.848**	0.520	0.368	0.466	**0.832**	0.574
02	**0.316**	**0.409**	**0.840**	0.618	**0.447**	0.506	0.840	0.518	**0.373**	**0.471**	0.808	0.571
03	0.313	0.413	0.768	**0.629**	0.439	**0.513**	0.744	**0.521**	0.370	0.466	0.784	**0.581**
File-2-File Evaluation with Manual Assessment												
Run IDs	**Japanese**				**Korean**				**Chinese**			
	MAP	**r-prec**	**p5**	**p50**	**MAP**	**r-prec**	**p5**	**p50**	**MAP**	**r-prec**	**p5**	**p50**
Best Scores	0.451	0.513	0.656	0.472	0.376	0.522	0.720	0.679	0.308	0.429	0.808	0.704
01	0.434	0.501	0.624	0.468	0.233	0.341	0.656	0.625	0.229	0.296	0.752	0.702
02	0.435	0.499	0.608	0.466	0.234	0.342	0.672	0.635	0.241	0.315	0.752	0.701
03	**0.451**	**0.513**	**0.656**	**0.472**	0.235	0.341	0.696	0.643	0.245	0.319	0.752	**0.704**
Anchor-2-File Evaluation with Manual Assessment												
Run IDs	**Japanese**				**Korean**				**Chinese**			
	MAP	**r-prec**	**p5**	**p50**	**MAP**	**r-prec**	**p5**	**p50**	**MAP**	**r-prec**	**p5**	**p50**
Best Scores	0.425	0.062	0.344	0.266	0.232	0.207	0.368	0.327	0.157	0.171	0.376	0.297
01	0.418	0.060	0.288	0.260	0.113	0.105	0.272	0.318	0.096	0.098	0.176	0.290
02	**0.425**	0.059	0.256	0.260	0.122	0.117	0.312	0.320	0.102	0.105	0.160	0.281
03	0.419	**0.062**	**0.344**	**0.266**	0.124	0.117	**0.368**	0.320	0.102	0.105	0.240	0.294

ation metric and do not have to be produced by one single system. In the file-2-file evaluation with Wikipedia ground truth, the HITS system is the top scoring system across all languages and evaluation metrics. The variations in the results across languages can be exclusively traced back to the mapping from English Wikipedia articles to the Wikipedia articles in the target language. The disambiguation process is exactly the same independent of the target language. The best scores are achieved by the setting from the run with ID 2, although the differences between the runs are small. In all runs, precision-at-n remains relatively stable until $n = 50$.

We achieve the top results in the file-2-file and anchor-2-file evaluation (MAP and r-prec) with manual assessment for Japanese. This shows that the multilingual index which we heavily rely on has the best coverage for Japanese. To improve the performance for Korean and Chinese, the mapping between English Wikipedia articles and those in the target languages has to be improved. As the results for Japanese show: the disambiguation system on which we focussed performs well. The run with ID 3 which includes features describing the specifity of the Wikipedia articles in the ranking process achieves the best results in this evaluation setting.

Analyzing the errors of HITS' disambiguation system indicates that problems arise when the candidate concepts for an anchor are closely related. In these cases, our features for disambiguation, which mainly describe how related a candidate concept is to its context, are not distinctive.

(4) ... in 1484 to serve the future *Holy Roman Emperor* Maximilian

(5) ... during the *administration* of Korea's first president

The anchor *Holy Roman Emperor* in (4) is for example disambiguated to the concept *Holy Roman Empire* instead of *Holy Roman Emperor*. In (5), the anchor *administration* is disambiguated as *President of the United States* instead of *Administration (government)*. In both examples, the entries in our lexicon are quite noisy. Errors such as the ones in these two examples could be partly avoided by using features measuring the distance between the anchor and the name of the respective candidate Wikipedia article. Moreover, we did not consider local context information such as selectional preferences.

Table 4: Settings for the different runs

Run IDs	Complete Run IDs	Setting
01	HITS_E2J_A2F_01 HITS_E2C_A2F_01 HITS_E2K_A2F_01	Threshold for ambiguity reduction step is set to $t = 0.4$
02	HITS_E2J_A2F_02 HITS_E2C_A2F_02 HITS_E2K_A2F_02	Threshold for ambiguity reduction step is set to $t = 0.1$
03	HITS_E2J_A2F_03 HITS_E2C_A2F_03 HITS_E2K_A2F_03	Threshold for ambiguity reduction step is set to $t = 0.1$. The ranking step uses features to describe specifity of the Wikipedia articles.

5. CONCLUSIONS

Our design decisions for building the HITS system have proven to be successful. We achieved the best performance in all task based evaluation metrics using Wikipedia ground truth and all the file-2-file and anchor-2-file evaluation metrics in the English-2-Japanese tasks. Our system builds on computational linguistics research in word sense disambiguation and implements a global disambiguation using a graph based approach. The core of the system is a multilingual index which works best for English to Japanese.

While the system performs well on English-2-Japanese, the performance on English-2-Chinese and English-2-Korean can be augmented by improving the mapping between English, Chinese and Korean Wikipedia articles.

6. ACKNOWLEDGMENTS.

This work has been partially funded by the European Commission through the CoSyne project FP7-ICT-4-248531 and the Klaus Tschira Foundation.

7. REFERENCES

[1] I. M. Bomze, M. Budinich, P. M. Pardalos, and M. Pelillo. The maximum clique problem. In D.-Z. Du and P. Pardalos, editors, *Handbook of Combinatorial Optimization*, pages 1–74. Kluwer Academic Publishers, Boston, Mass., 1999.

[2] R. Bunescu and M. Paşca. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics,* Trento, Italy, 3–7 April 2006, pages 9–16, 2006.

[3] A. Csomai and R. Mihalcea. Linking documents to encyclopedic knowledge. *IEEE Intelligent Systems*, 23(5):34–41, 2008.

[4] S. 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 Language Learning,* Prague, Czech Republic, 28–30 June 2007, pages 708–716, 2007.

[5] P. Ferragina and U. Scaiella. TAGME: On-the-fly annotation of short text fragments (by Wikipedia entities). In *Proceedings of the ACM 19th Conference on Information and Knowledge Management (CIKM 2010),* Toronto, Ont., Canada, 26–30 October 2010, pages 1625–1628, 2010.

[6] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence,* Hyderabad, India, 6–12 January 2007, pages 1606–1611, 2007.

[7] S. Kulkarni, A. Singh, G. Ramakrishnan, and S. Chakrabarti. Collective annotation of Wikipedia entities in web text. In *Proceedings of the 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining,* Paris, France, 28 June – 1 July 2009, pages 457–466, 2009.

[8] O. Medelyan, I. H. Witten, and D. Milne. Topic indexing with Wikipedia. In *Proceedings of the Workshop on Wikipedia and Artificial Intelligence: An Evolving Synergy at AAAI-08,* Chicago, Ill., 13 July 2008, pages 19–24, 2008.

[9] D. Milne and I. H. Witten. An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In *Proceedings of the Workshop on Wikipedia and Artificial Intelligence: An Evolving Synergy at AAAI-08,* Chicago, Ill., 13 July 2008, pages 25–30, 2008.

[10] D. Milne and I. H. Witten. Learning to link with Wikipedia. In *Proceedings of the ACM 17th Conference on Information and Knowledge Management (CIKM 2008),* Napa Valley, Cal., USA, 26–30 October 2008, pages 1046–1055, 2008.

[11] V. Nastase, M. Strube, B. Börschinger, C. Zirn, and A. Elghafari. WikiNet: A very large scale multi-lingual concept network. In *Proceedings of the 7th International Conference on Language Resources and Evaluation,* La Valetta, Malta, 17–23 May 2010, 2010.

[12] M. E. Newman. *Networks: An Introduction*. Oxford University Press, New York, N.Y., 2010.

[13] S. P. Ponzetto and M. Strube. Knowledge derived from Wikipedia for computing semantic relatedness. *Journal of Artificial Intelligence Research*, 30:181–212, 2007.

[14] W. Pullan. Approximating the maximum vertex/edge weighted clique using local search. *Journal of Heuristics*, 14(2):117–134, 2008.

[15] L. Ratinov, D. Roth, D. Downey, and M. Anderson. Local and global algorithms for disambiguation to Wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics,* Portland, Oreg., USA, 19–24 June 2011, pages 1375–1384, 2011.

[16] R. Schenkel, F. M. Suchanek, and G. Kasneci. YAWN: A semantically annotated Wikipedia XML corpus. In *Proceedings of Datenbanksysteme in Business, Technologie und Web (BTW 2007), 12. Fachtagung des GI-Fachbereichs "Datenbanken und Informationssysteme,* Aachen, Germany, 7-9 March 2007, pages 277–291, 2007.

[17] N. Seco, T. Veale, and J. Hayes. An intrinsic information content metric for semantic similarity in WordNet. In *Proceedings of the 16th European Conference on Artificial Intelligence,* Valencia, Spain, 23–27 August 2004, pages 1089–1090, 2004.

[18] L.-X. Tang, S. Geva, A. Trotman, Y. Xu, and K. Itakura. Overview of the NTCIR-9 crosslink task: Cross-lingual link discovery. In *Proceedings of the 9th NTCIR Workshop Meeting,* Tokyo, Japan, 6-9 December 2011, 2011. To appear.

[19] D. Y. Turdakov and S. D. Lizorkin. HMM expanded to multiple interleaved chains as a model for word sense disambiguation. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation,* Hong Kong, China, 3–5 December 2009, 2009.

[20] W. Wentland, J. Knopp, C. Silberer, and M. 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,* Marrakech, Morocco, 26 May – 1 June 2008, 2008.

[21] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco, Cal., 2nd edition, 2005.

[22] Y. Zhou. Resolving surface forms to Wikipedia topics. In *Proceedings of the 23rd International Conference on Computational Linguistics,* Beijing, China, 23–27 August 2010, pages 1335–1343, 2010.