Semantic Extraction of Key Figures and Their Properties From Tax Legal Texts Using Neural Models

Daniel Steinigen[1], Marcin Namysl[2], Markus Hepperle[3], Jan Krekeler[3] and Susanne Landgraf[4]

[1]Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS, Schloss Birlinghoven 1, Sankt Augustin, 53757, Germany
[3]Bucerius Law School, Jungiusstraße 6, Hamburg, 20355, Germany
[4]Federal Ministry of Finance, Wilhelmstraße 97, Berlin, 10117, Germany

Abstract

Applying information extraction to legislative texts is a challenging task that requires a specification to distinguish the relevant parts from the less relevant parts of the text. Moreover, there is still a lack of appropriate language- and domain-specific data in the field of information extraction. This work investigates the extraction and modeling of key figures from legal texts. We introduce a universally applicable annotation scheme together with a semantic model for key figures and their logically connected properties in legal texts. Moreover, we release *KeyFiTax*, a dataset with key figures based on paragraphs of German tax acts manually annotated by tax experts together with a knowledge graph populated from these paragraphs based on our semantic model. Using our dataset, we also evaluate and compare state-of-the-art entity extraction models in terms of long entity spans and low-resource data. Furthermore, we present a transformer-based approach for relation extraction using *entity markers* to obtain a logical formulation of the key figures. Finally, we introduce *task triggers* for training a combined resource-efficient entity and relation extraction model. We make our dataset together with the semantic model and the knowledge graph, as well as the implementation of the entity and relation extraction approaches investigated in this work public.

Keywords

information extraction, entity extraction, relation extraction, ontologies, knowledge graphs, transformers, language models, German datasets, legal texts, tax key figures

1. Introduction

Key figures represent a central component in legal texts of tax laws. They are crucial for applying laws and are an important criterion in the amendment of laws. Such key figures are, e.g., *Entfernungspauschale* 'distance allowance', *Kinderfreibetrag* 'child tax-free allowance' or *Werbungskostenpauschale* 'flat-rate income-related expenses allowance'.

Changing key figures in the tax laws directly affects the resulting tax revenue. An example would be increasing the commuter allowance to 50 cents per km. In terms of estimating the impact of a change in the law, a model can be used to simulate what effect an adjustment of the key figures will have on the specific tax forecast. To facilitate this, in this paper we propose an approach based on information extraction and semantic technologies. For this it is first necessary to recognize and extract the key figures with their logically connected properties and rules from legal texts. This task requires an automatic understanding of the legal texts and recognizing the relevant information within the text. Then it is necessary to semantically model the extracted information using a specific ontology and populate a Knowledge Graph (KG) out of this information. This then allows to compare the KG's of existing and new law texts to identify legislative changes. In this paper we focus on the information extraction part and the semantic modeling part. We leave the differential analysis and the prediction of the impact on tax revenue for future work.

We use natural language processing (NLP) and ma-

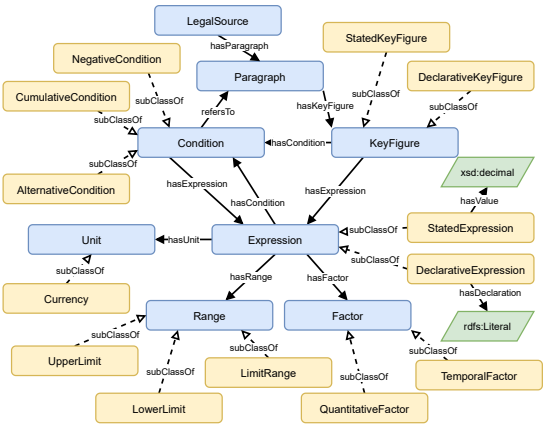

Figure 1: Ontology for semantic modeling of key figures and their logically connected properties in legal texts

Proceedings of the Sixth Workshop on Automated Semantic Analysis of Information in Legal Text (ASAIL 2023), June 23, 2023, Braga, Portugal.
Daniel.Steinigen@iais.fraunhofer.de (D. Steinigen);
Marcin.Namysl@iais.fraunhofer.de (M. Namysl);
Markus.Hepperle@law-school.de (M. Hepperle);
Jan.Krekeler@law-school.de (J. Krekeler)
0000-0001-9039-2965 (D. Steinigen); 0000-0001-7066-1726 (M. Namysl)

Figure 2: Two excerpts from paragraphs annotated according to our developed annotation scheme

chine learning (ML) approaches to extract key figures from legal texts. Specifically, we consider this problem a token-level classification task, known as *sequence labeling*. With this approach, each token of a text is classified according to the predefined categories, whereby tokens not assigned to any class are labeled with zeros [1]. More precisely, this can also be interpreted as an entity extraction task in which individual entities can span over many words or tokens. Entity extraction is widely used in the research area of information extraction (IE) and has also been applied in the legal domain [2].

We face several challenges in applying standard entity extraction approaches in our work. Since we focus on German tax legal texts, we have both language- and domain-specific data. It means we are in a low-resource domain and have to deal with limited training data. Moreover, the entities can span over many tokens, making it harder for the models to recognize the complete entities. Furthermore, not all numeric currency values are directly relevant to key figures. Therefore, the model must learn the text semantics and what specific tokens refer to in order to extract the relevant values.

For obtaining a logical formulation of the key figures, it is necessary to extract the key figures represented by their entities and the relations between them. To address this, we also consider relation extraction approaches in our work. To facilitate resource-efficient training and to get more benefit from the limited amount of available training data, training a combined model for both entity and relation extraction is reasonable.

As a prerequisite to training models for the automatic extraction of key figures, we also introduce an annotation scheme together with a semantic model for key figures in legal texts. A variety of approaches, ontologies, and knowledge graphs already exist for semantic modeling of legal texts. *LegalRuleML* by Palmirani et al. [3] is intended to model legal rules and to connect between legal sources and metadata of the rules. They also introduce a Metamodel with defined nodes (classes) and edges (properties) to expose the LegalRuleML Metadata as linked data. Moreno Schneider et al. [4] propose a Legal Knowledge Graph that integrates and links heterogeneous compliance data sources including legislation, case law, regulations, standards, and private contracts. Holzenberger and Van Durm [5] investigated the performance of natural language understanding approaches on statutory reasoning by introducing the *SARA* dataset, which consists among other of extracted arguments and a graph-based representation of those arguments. Nevertheless, these approaches are either too general and generic or too specifically modeled for a particular problem to fit our use case of modeling key figures. Therefore, we propose a new semantic model tailored to our use case that models the key figures with their properties in detail. The authors of this paper are a diverse team of NLP and ML experts and tax experts. In interdisciplinary cooperation, we have developed an annotation scheme and a semantic model in an iterative process, which contains the classes and properties required for the complete specification of the key figures.

There are various challenges when annotating the key figures. Since legal texts can be structured in a complex way, the goal is to find a universally applicable annotation schema. Furthermore, most key figures contain not just a single value but different values that apply under different conditions. Using the created annotation scheme, we generated a manually annotated gold standard dataset based on paragraphs of German tax laws. This dataset is the basis for training and evaluating different state-of-the-art information extraction models. Figure 2 shows two examples of annotated paragraphs with distinct categories or entity types.

By applying our information extraction models and our semantic model, the adjusted key figures will be extracted and semantically modeled when the legal texts have changed so that they can be taken into account in the tax forecast. In summary, the contributions of this paper are as follows:

- An annotation scheme together with a semantic model for key figures in legal texts
- A dataset consisting of paragraphs of German tax laws with annotated key figures and a knowledge graph populated with these key figures
- Evaluation and comparison of state-of-the-art entity extraction models in terms of long entity spans and low-resource data utilizing the proposed dataset
- A transformer-based approach for a combined resource-efficient extraction of entities and relations from legal data.

2. Semantic Model and Dataset

2.1. Data Sources and Data Selection

The initial data basis for generating the annotated dataset is legal texts in the German language. For this purpose, we took advantage of the publicly accessible website of the German Federal Ministry of Justice and the Federal Office of Justice[1], which contains the current German laws and legal regulations. These legal texts are available in various data formats, such as XML, PDF, or HTML. For our purpose, we use the XML files and automatically extract the contained legal paragraphs.

In accordance with the overall aim of providing a model for determining the impact of legislative change on tax revenues, we select on a primary step the relevant German tax laws, notably the Fiscal Code (Abgabenordnung), the Income Tax Act (Einkommensteuergesetz), Corporate Tax Act (Körperschaftsteuergesetz), Inheritance Tax Act (Erbschaft- und Schenkungsteuergesetz) and further tax acts regulating German direct and indirect taxes. To generate a larger dataset, we also considered further tax acts from other jurisdictions in the German language, such as the Austrian or the Swiss, but gave up on this due to the inconsistent and, therefore, harmful use of the same key figures in a differing meaning or different key figures in the same meaning as the key figure from the German jurisdiction.

In the second step, we determine the relevant sections and paragraphs of the selected acts. To this end, we ask which rules directly impact the tax revenues and have not only a serving or systematizing function. Thereto we select these sections and paragraphs, which contain a key figure and a corresponding value and unit, which are the essential and mandatory components of the relevant key figures, whereas the other categories are optional. The categories are described in detail in the next section.

2.2. Semantic Model

We introduce our annotation scheme and our semantic model for creating the dataset with different semantic categories for the key figures. The goal is to provide a comprehensive specification of the key figures so that they can be used independently of the legal text for downstream applications, such as tax forecasts. The annotation scheme and the semantic model should be universally applicable to legal texts, which can be structured in various complex ways. We identified the semantic categories in an iterative process by analyzing different paragraphs of tax acts and revising our annotation scheme continuously.

First, we introduce the category for the *key figure* itself as a central category, which is specified by containing one or more values that have an impact on tax revenue. The annotation can be considered as the name or label for the key figure. It corresponds to a text phrase or word that describes this key figure. Figure 2 shows, for example, the annotation of the key figures *distance allowance* and *child allowance*. Then, since every key figure we consider here should have at least one or more values, there is the category for these values that we call *expression* of the key figure. These are numerical values or terms to which the key figures refer, such as the values *0.30* or *4 500* in Figure 2. The expressions can be specified in certain units, so there is a category for *units*. In the case of monetary amounts, which often appear in tax acts, the unit is in most cases a currency, such as *Euro*.

With these three categories, simple key figures can already be specified. However, while analyzing the legal texts, we found that the key figures can also be structured much more complexly. Thus, most key figures contain not only a single expression but different expressions that apply under different conditions, and there are also preconditions for specific key figures. For this purpose, we introduce the category *condition*. It includes spans of text over several words with conditions that apply to a key figure or for which a key figure has specific expressions. An example is the commuter allowance, which amounts to 0.30 euros up to 20 kilometers driven and increases to 0.35 euros from kilometer 21. Another example is given in Figure 2, where it is shown that the child allowance can have different expressions resp. values depending on the number of children, which is the condition there.

We also found that there can be different types of conditions, namely *negative*, *alternative*, or *cumulative conditions*. An example of a negative condition can be found in section 24 sentence 2 of the Corporate Tax Act. The provision stipulates that the allowance for corporate tax subjects, as regulated in sentence 1, does not apply to the type of subjects specified in number 1 to 3 of the provision. Alternative conditions are, for instance, used in section 10b para. 1 Sentence 8 of the Income Tax Act. The sentence regulates that certain membership fees cannot be deducted in case they are paid to corporations serving certain in number 1 to 5 specified purposes. The deduction prohibition already applies, if only one of these numbers is fulfilled, as indicated by the word *or* between the ultimate and the penultimate number. Section 10b para. 1a sentence 1 contains one of many examples of cumulative conditions, where donations into the assets of a foundation are only declared deductible, if they meet the requirements of a donation into the assets of a foundation, the provisions of para. 1 sentence 2 to 6 are fulfilled, and an application has been filed.

Another point to consider when describing key figures is that the expressions are not always just fixed values but can also define a range in which a key figure applies. This

[1] https://www.gesetze-im-internet.de

Table 1
Semantic categories with some sample formulations and their English translations

Category	Examples in German	English Translations
Key figure (stated)	"Pauschbeträge für Werbungskosten", "Entfernungspauschale"	"Lump sums for advertising expenses", "distance allowance"
Key figure (declarative)	"Steuerabzug von den nach Abzug der Betriebsausgaben oder Werbungskosten verbleibenden Einnahmen"	"Tax deduction from the income remaining after deduction of operating expenses or income-related expenses"
Expression (stated)	"0.35", "2 Millionen", "30 Prozent"	"0.35", "2 millions", "30 percent"
Expression (declarative)	"10 Prozent der gesamten Einkünfte der ausländischen Gesellschaft"	"10 percent of the total income of the foreign company"
Unit	"Euro", "EUR", "€"	"Euro", "EUR", "€"
Condition	"Einnahmen aus nichtselbständiger Arbeit"	"Income from non-employee work"
Range	"insgesamt bis zu", "von mindestens"	"in total up to", "of at least"
Factor	"pro Kilometer", "monatlich"	"per kilometer", "monthly"

is covered by the *range* category. The range is an indicator for the area in which an expression is valid. This area can be defined by either an upper limit, a lower limit or some limit range. Figure 2 shows an example of an upper limit *"at most"* and a lower limit *"an amount greater than"*. In addition, there is also weighting of the expressions, which we call *factors*. This category characterizes the factor that must be considered for a expression and indicates what the expression refers to. These factors can be further divided into temporal factors, which refer to periods of time, such as months or years, and quantitative factors, which refer to some absolute amount. For example, the paragraph in Figure 2 includes a temporal factor *"per calendar year"* and a quantitative factor *"for each full kilometer"*.

Furthermore, we found that not all key figures have their expressions explicitly mentioned as such in the legal texts. It means that the key figures sometimes cannot be recognized as distinct mentions of a short sequence of words, and expressions do not always occur as easily recognizable numerical values. Instead, the key figures and expressions can also be implicitly described in the legal texts using long phrases in a declarative manner. To tackle this, we have two additional categories for the declarative phrases of the key figures and expressions, called *declarative key figures* and *declarative expressions*. For the cases where the key figures and expressions are explicitly mentioned, we use the categories *stated key figure* and *stated expression*. Table 1 shows all introduced semantic categories with some sample formulations and their English translations.

To assign the annotations created according to the semantic categories to each other in order to obtain a logical formulation of the key figures, we also introduce relation types between the categories. This is also particularly important, since a single paragraph may contain multiple key figures with the associated other categories. Based on the defined semantic categories and relation types, we build a semantic model in the form of an ontology as shown in Figure 1 using the RDF Schema[2] vocabulary. The semantic categories become the classes and the relations become the properties of this ontology, which also define the permissible properties between these classes. For the class expression, we have also defined data properties for storing the numeric values if they are explicitly specified or the phrases for the declarative expressions. This model allows the assignment of the key figures to the associated conditions and expressions during annotation. Noteworthy are the properties *hasCondition* and *hasExpression* since they can be applied to two different classes as a head. When considering *conditions*, these can apply directly to certain *key figures* or define the validity of different *expressions*. On the other hand, *expressions* can be derived directly from a *key figure* or can also be part of a *condition*.

Furthermore, we introduce the relation *join* to link related annotations from the same semantic category since there are cases where a single entity is spread across multiple annotations. Beyond the key figures, we also model the pargraphs that contain the key figures and the legal sources, in our case the tax acts that consist of the parapgraphs. In addition, since conditions can be expressed not only by natural text, but also depend other paragraphs, we also introduce a property *referTo* between condition and paragraph.

2.3. Annotation Rules and Dataset Acquisition

Given the developed annotation schema and the collected data sources, the next step is to annotate the legal texts and build up the dataset. For the further procedure of annotating the dataset and applying the information extraction models, we refer to the *semantic catgories* or *classes* as *entities* and the *properties* as *relations*. We first used the selected paragraphs from Section 2.1 and performed a simple pre-annotation task. Using rule-based

[2] https://www.w3.org/TR/rdf-schema/

approaches and pattern matching, we automatically enriched the paragraphs with annotations for the *expression* and *unit* categories. The annotators reviewed these pre-annotations and corrected, removed, or complemented them as necessary. For storing the pre-annotated data, we have chosen the CAS format serialized as an XMI file. It allows us to import the data directly into the annotation tool. For manual annotation of the texts, we use the INCEpTION tool[3] [6] as it has an intuitive graphical user interface and can be configured well for specific annotation tasks.

Furthermore, we defined a set of annotation rules. We only allow complete words to be annotated and not parts of words. We do not allow multi-label annotation except for the *conditions* category, which means that each token can only be labeled with one of the defined semantic categories. *Conditions* are an exception to this rule. Each token already labeled as a *condition* can also have a label of another category because *conditions* can also represent a *key figure* concurrently, and *conditions* themselves can contain *expressions*. For example, section 10 para. 1a sentence 1 number 1 of the Income tax act contains the key figure of maintenance payments to the divorced or permanently separated spouse who is subject to unlimited income tax liability, which is a condition of this key figure. This is because the key figure and its expression only apply if the maintenance payment, as defined elsewhere (in the German Civil Code) but referenced here, is paid.

We also found that besides different types, the conditions can also have different formats. Considering the length, some conditions that span only a few words, and others might span entire sentences. Here we do not limit the length of the conditions and allow arbitrary long phrases. The same applies to the categories *declarative key figure* and *declarative expression*.

For our annotation task we simplify for now the issue that there are different condition types, and do not distinguish these types during annotation. We define that cumulative conditions are labeled contiguously and that alternative conditions are labeled separately as long as they do not have a common beginning or end of sentence. In addition, the relations between the entities are also annotated. However, the relations are only allowed between certain entity types, in a defined direction. This annotation was done in accordance with the classes and properties defined in the ontology in Figure 1.

The data was annotated by tax experts who coauthored this paper in an iterative process. In this process, we also continuously developed the annotation scheme together with the semantic model. The first semantic model was more restrictive and as it progressed we allowed more relations when it was necessary. In addition to the specifications already mentioned, there were other aspects and challenges to be considered during the annotation. The general challenge is the complexity of the German tax regulations, which are often long, convoluted, and contain references to other provisions. Hence, compromises were often necessary between annotation as accurately as possible and managing the complexity of annotations that would otherwise result in specifying rules that affect only a small number of tokens. Because the dataset is of a manageable size, the annotation agreement was that the annotation is done piecewise by both commenters simultaneously. Anomalies and deviations were then discussed together with the NLP engineers and the annotation scheme was readjusted if necessary.

2.4. Dataset Statistics

The generated dataset includes 106 annotated paragraphs from 14 different German tax acts. Table 2 show the statistics of the generated dataset with the number of annotated instances and the token sequence length for each category. It shows that the dataset contains 157 annotations of *key figures*, with the corresponding additional categories. The statistics also illustrate that the annotations for categories *condition*, *declarative key figure*, and *declarative expression* contain very long token sequences. We further populated a KG out of this annotated dataset using the defined semantic model from Section Section 2.2. The annotated dataset, as well as the KG and the list of tax acts of which paragraphs are included in the dataset, have been made publicly available and can be found in the project repository.

Table 2
Statistics of the entities and relations in our dataset. *No.* is the number of annotated instances and *Tok.* the mean number of tokens for each category.

Entity Type	No.	Tok.	Relation Type	No.
Key figure (stated)	129	4	hasKeyFigure	157
Expression (stated)	295	2	hasExpression	319
Unit	284	1	hasUnit	279
Condition	491	14	hasCondition	399
Range	75	2	hasRange	75
Factor	97	11	hasFactor	137
Key figure (declarative)	28	14	hasParagraph	106
Expression (declarative)	32	6	join	139

[3] https://inception-project.github.io/

3. Approaches for Key Figure Extraction

Given the dataset described in Section 2, the goal is to automatically extract the key figures specified by their semantic types from the legal texts. We address this problem by employing entity extraction approaches. In the entity extraction task, each token of a text is assigned a label according to some predefined categories, whereby tokens not assigned to any category are labeled with zeros. The individual entities can then span over a large number of tokens. Based on this, ML-based classification models can be trained to classify each token. Ideally, the model memorizes the examples seen during training and tries to generalize to unseen examples.

3.1. Approaches from NLP libraries

In our work, we consider and compare different approaches for entity extraction. First, we investigate the approaches of two well-known NLP libraries *spaCy* and *RASA*. For *spaCy*, we take advantage of the provided predefined pipelines for training named entity recognition (NER) models[4]. We used the recommended settings and adjusted the hyperparameters for our use case, as shown in Table 7. From *RASA*, we use an entity extraction approach based on a conditional random field (CRF) model[5]. This model utilizes the *sklearn-crfsuite*[6] and uses features of the words (e.g., capitalization, part-of-speech tagging) and their context to assign probabilities to certain entity classes.

3.2. Transformer Models for Entity Extraction

We also consider transformer-based approaches as we investigate the low-resource scenario and have to cope with long entity spans. Transformer architecture aims to solve sequence-to-sequence tasks while being able to consider long-distance dependencies across several words in a sentence by employing the attention mechanism [7]. Transformer-based language models can be pre-trained on large text corpora, allowing them to understand the contextual relationships between individual words and sentences. Considering the entity extraction task, we choose models that utilize the encoder part of the transformer architecture. These models provide an encoded representation of the input sentences. We use a final classification layer to classify the sentence tokens according to our annotation scheme.

For our work, we select relevant models pre-trained on German text data. First, we consider the *German BERT* model and the *GBERT* and *GElectra* models by Chan et al. [8], which, in addition to Wikipedia- and news articles, is also pre-trained on 2.4GB of German legal texts from Open Legal Data[7] [9]. We also consider a multilingual language model *XLM-RoBERTa* [10], which is pre-trained on 2.5 TB of data from 100 different languages, including about 100 GB of German texts.

In order to face the challenge of long input sequences due to the long paragraphs legal texts can have, we also consider the *Longformer* model by Beltagy et al. [11]. In contrast to the other models, which only allow a maximum length of 512 tokens as input, this model allows up to 4096 tokens. Specifically, we use the *XLM-R Longformer* model by Sagen [12][8]. This is an *XLM-RoBERTa* model that has been extended to allow sequence lengths up to 4096 tokens using the *Longformer* pre-training scheme.

3.3. Relation Extraction

As described in Section 2, our goal is to automatically extract key figures represented by their entities and the relations between them to obtain the logical formulation of key figures. We employ a relation extraction approach to classify the relationship between the entities. Table 2 lists the relations in our dataset. Note that a simple rule-based assignment of the relation type based on the entity types according to the ontology in Figure 1 is not straightforward as the relationship may or may not exist depending on many other factors. Therefore, we apply ML-based approaches to this task.

We adopt a transformer-based approach inspired by Zhou and Chen [13] and introduce *typed entity markers* to the input text before feeding it into the model. First, we add *special tokens* into the vocabulary of the model and use them to enclose subject and object entities within the input paragraph: `[SUB]`, `[/SUB]`, `[OBJ]`, `[/OBJ]`. In addition to the subject and object, we also mark the type of entities in the input text by using additional special tokens for each entity type, which provides the neural network with prior knowledge that facilitates the learning process.

Multiple training samples are generated for each input paragraph depending on the number of entities contained in that paragraph. For each sample, we mark one entity as a subject and all other entities as objects. Similar to the sequence labeling approach (Section 3.2), we feed the text with marked entities to the encoder to obtain a token-level representation of the input. Then, we apply a classification layer to classify the relations between the subject and objects. We label each `[OBJ]` token with the

[4] https://spacy.io/usage/training/
[5] https://rasa.com/docs/rasa/components/#crfentityextractor
[6] https://sklearn-crfsuite.readthedocs.io/en/latest/
[7] http://openlegaldata.io/research/2019/02/19/court-decision-dataset.html
[8] https://github.com/MarkusSagen/Master-Thesis-Multilingual-Longformer

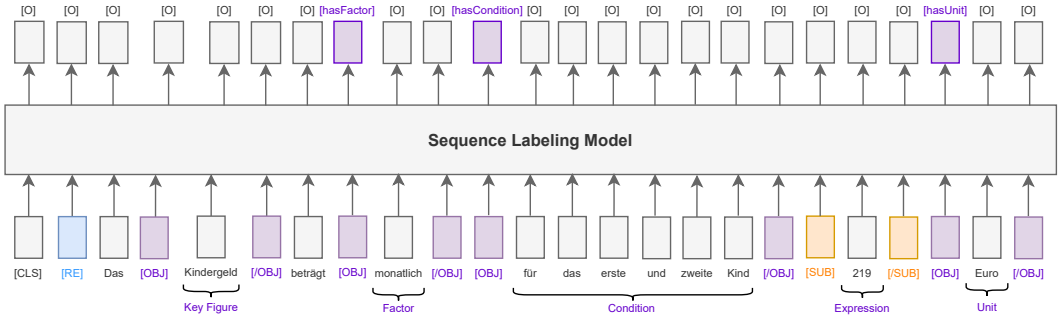

Figure 3: Excerpt from a paragraph with marked entities, labeled relations and trigger token for relation extraction accordant our proposed approach

relation type between the subject entity and the corresponding object entity. Entities with no relation to the subject entity are labeled with zeros.

3.4. Joint Entity and Relation Extraction

Extending the approach from Section 3.3 further, it is possible to use the same network architecture to train a combined entity and relation extraction model. To this end, we introduce new special tokens called *task triggers* to distinguish the entity and relation extraction task: `[EE]` and `[RE]`, respectively. We insert these tokens at the beginning of each paragraph right after the `[CLS]` token.

Moreover, since the condition class may overlap with other classes in our dataset, we employ *task triggers* to distinguish between groups of entities by defining additional triggers for each group. It allows us to separate entities into groups of types with non-overlapping annotations. Specifically, we have one entity group for conditions, marked with `[GRP-1]`, and one group for the remaining entity types, marked with `[GRP-2]`. Considering that we have two entity groups, this gives us two training samples for entity extraction and multiple samples (depending on the number of entities) for relation extraction for each paragraph. By executing multiple forward passes on a single token classification model, we can recognize entities with overlapping annotations as well as the relations between these entities.

One advantage of this approach is that we do not need to train separate models for the different entity groups and for relation extraction, which saves computational resources for training and memory resources for inference. Another advantage is that we get a larger number and variety of samples for training the model and thus more benefit from the limited training data available. Figure 3 shows an example excerpt of a paragraph with the marked entities and the labeled relations. A detailed overview of all generated training samples for this excerpt can be found in the project repository.

4. Experimental Evaluation

4.1. Comparison of Approaches for Entity Extraction From Legal Data

In this experiment, we evaluate the entity extraction approaches described in Section 3 on the dataset introduced in Section 2. For this purpose, we only use the superclasses *condition*, *unit*, *range* and *factor* of our semantic model and do not distinguish into the subclasses. However, for the *key figure* and *expressions* classes we retain the distinction between the *stated* and *declarative* subclasses.

4.1.1. Experimental Setup

Data Split and Data Partition We use different strategies for splitting the data. For evaluating the different types of transformer models described in Section 3.2 and finding the best-performing model on our dataset, we randomly split the data into fixed training (80%) and evaluation (20%) subsets. This results in 85 paragraphs for training and 21 for evaluation. For the condition class, which is trained separately, there are 73 paragraphs for training and 18 for evaluation. This allows us to identify the most suitable models in less time and with less computational effort compared to the more complex evaluation approach we used afterward.

In the next step, we select the best-performing transformer model and compare it with the other approaches described in Section 3.1 using k-fold cross-validation. This validation technique is particularly suitable for the low-resource scenario considered here, as it reduces the influence of the distribution of data across the training and test splits on the evaluation results of the models. We choose k= 5 and randomly divide the dataset into five equal-sized subsets. In each iteration, one subset is retained as the data used for testing the model, and the remaining four subsets are used as training data. Thus, each subset is used once for evaluation and four times

for training the model. The results are then averaged to produce the final scores.

Training Setup As the annotations for the *condition* class may overlap with other annotations, we train two separate models — one for the recognition of the *condition* type and the other for the recognition of the remaining entity types. We train the transformer model over 200 epochs with a batch size of 8 and a learning rate of 1×10^{-5}. All other relevant hyperparameters and the configuration files used for the other approaches are documented in the project repository.

Evaluation Metric For each entity type individually, we report the token-level micro-averaged F_1 *score* on the test set as the evaluation metric in the charts. We also provide the macro-averaged F_1 *score* over all classes as a tabular overview. For k-fold cross-validation, we report the average F_1 *score* achieved over all five training runs.

4.1.2. Results and Discussion

Transformer Models The evaluation results for comparing different pre-trained transformer models are presented in Table 3 as a summary overview. The detailed performance of the evaluated models per class is visualized in the project repository. The results show that the GBERT and XLM-RoBERTa models outperform other models for the *declarative expression* class. The best-performing Transformer model is XLM-RoBERTa$_{\text{LARGE}}$ with a F_1 *score* of 56.8 %.

Model comparison By choosing XLM-RoBERTa$_{\text{LARGE}}$, we perform a cross-validation of this model and the spaCy-NER and RASA-CRF approaches. Figure 4 present the results of this experiment.

In the case of the *unit* class, all models achieved high F_1 scores. Unsurprisingly, the instances of this class are single-token entities (e.g., *Euro*, *EUR*) that only pose a few challenges to the examined models. Similarly, the scores for the *stated expression* class were also high.

The *Range* and *Factor* classes were recognized relatively well, especially by XLM-RoBERTa$_{\text{LARGE}}$ and in the case of the *Factor* class also by spaCy-NER. Note that these two classes have three times fewer samples than in the case of the *expression* and *unit* types. Despite a lower number of examples, similar scores are achieved on the *declarative expression* class by XLM-RoBERTa$_{\text{LARGE}}$.

All models, except XLM-RoBERTa$_{\text{LARGE}}$, perform relatively poorly on the *key figure* class. Interestingly, the variance of the results for this class is relatively large: RASA-CRF achieves only 0.16 F_1 score and, in contrast, XLM-RoBERTa$_{\text{LARGE}}$ exhibits three times better score.

For the *declarative key figure* class, the performance of every model examined in our experiment is the worst. We believe that it is due to the complexity of this class and the low number of instances in the data (see *num. samples* and *max. length* plots in Figure 4, respectively).

Despite a large number of available samples, the score on the *condition* class is also low for spaCy-NER and RASA-CRF, but acceptable for XLM-RoBERTa$_{\text{LARGE}}$. We believe that the length and the complexity of this class could cause this. Note that the longest instances of this class have over 100 tokens. Moreover, the concept of a *condition* is not so strictly defined, as, e.g., *expression*, *unit*, or *factor*.

Looking at the overall performance across all classes, XLM-RoBERTa$_{\text{LARGE}}$ clearly scores the best with a macro-averaged F_1 *score* of 60.9 %. SpaCy-NER and RASA-CRF perform comparably in terms of overall performance but are still about 15 % behind XLM-RoBERTa$_{\text{LARGE}}$.

Table 3
Results of the entity extraction models presented in Section 4.1. For each model we present the macro-averaged F_1 *score* over all classes

Model	F_1 (in %)
GBERT$_{\text{BASE}}$	53.97
GBERT$_{\text{LARGE}}$	52.59
GElectra$_{\text{BASE}}$	44.44
GElectra$_{\text{LARGE}}$	44.29
Longformer	38.88
XLM-RoBERTa$_{\text{BASE}}$	55.20
XLM-RoBERTa$_{\text{LARGE}}$	56.80
spaCy-NER (cross-validated)	45.78
RASA-CRF (cross-validated)	44.10
XLM-RoBERTa$_{\text{LARGE}}$ (cross-validated)	60.91
XLM-RoBERTa$_{\text{LARGE}}$-Triggers (cross-validated)	58.78

4.2. Combined Extraction of Entities and Relations From Legal Data

In this experiment, we evaluate the approach described in Section 3.4 for combined entity and relation extraction on the dataset introduced in Section 2. We use the same classes as in Section 4.1.

4.2.1. Experimental Setup

Training Setup We select the XLM-RoBERTa$_{\text{LARGE}}$ model for this experiment as its results in Section 4.1 were the most consistent among the examined models. Using the approach described in Section 3.3, we train one model for extracting the two groups of entities and the relations.

Dataset We expand our training data according to Section 3.3. For each record, we create one training sample

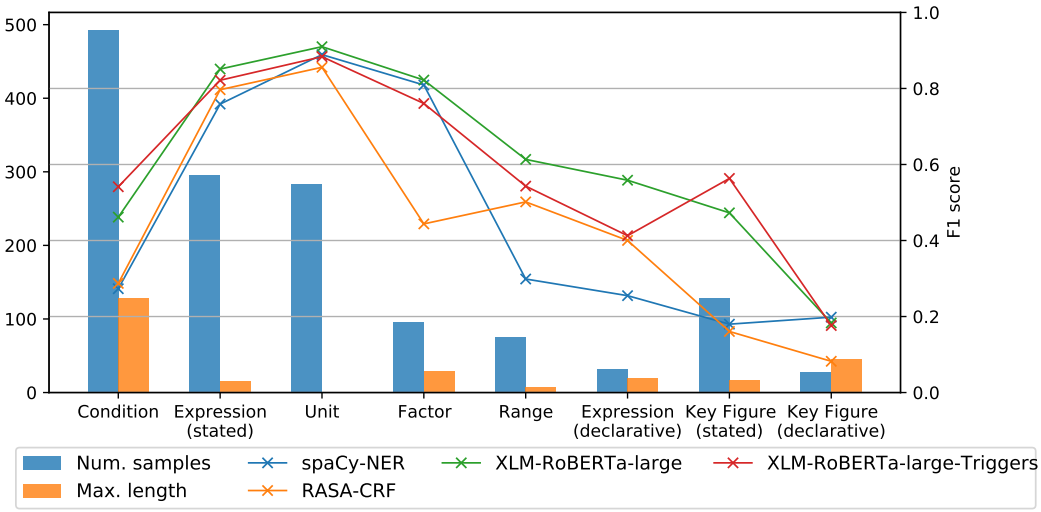

Figure 4: Results of the experiment presented in Section 4.1. For each entity type, we present the averaged F_1 scores for cross-validation, the total number of samples, and the maximum number of tokens of a single instance.

for each entity group and one training sample for each possible subject entity containing the entity markers for relation extraction. Then, analogous to Section 4.1, we apply cross-validation to evaluate the model's performance.

Table 4
Results for relation extraction presented in Section 4.2. We present the F_1 *score* for each class as well as the macro-averaged F_1 *score* over all classes

Relation	F_1 (in %)
hasCondition	62.99
hasExpression	72.54
hasUnit	97.37
hasFactor	76.60
hasRange	85.88
join	68.66
Macro-averaged	77.34

4.2.2. Results and Discussion

Entity Extraction The results of this experiment are presented in Figure 4 and Table 3, named as XLM-RoBERTa$_{\text{LARGE}}$-Triggers, for comparison with the other models. The evaluation result shows that the jointly trained model can achieve comparable performance for entity extraction as the XLM-RoBERTa$_{\text{LARGE}}$ models trained separately for conditions and other entities. Even though the jointly trained model slightly underperforms on the classes *factor*, *range* and *declarative expression* compared to the separately trained models, XLM-RoBERTa$_{\text{LARGE}}$-Triggers achieves better performance on the most relevant class *key figure*. Moreover, it performs better on the most complex class *condition*.

Relation Extraction The performance of this model in the relation extraction task is presented in Table 4. The result shows that the F_1 *scores* for all relation types are above 0.6. Especially for relations *hasUnit*, *hasRange*, *hasFactor* and *hasExpression* the F_1 *scores* are high. The model recognized the relationship between expressions and units almost perfectly.

5. Related Work

5.1. NLP datasets in Legal Domain

Chalkidis et al. [14] provide a dataset for entity recognition consisting of 3,500 English contracts manually annotated with 11 entity types (party name, termination date, jurisdiction, etc.). Chalkidis et al. [15] release a multi-label text classification dataset based on EUR-LEX portal[9]. Leitner et al. [16] develop a dataset consisting of German court decisions annotated with 19 entity types (person, judge, lawyer, ordinance, court decision, etc.) and they examine, among others, CRF's for entity extraction. Glaser et al. [17] introduce a dataset of 100k German court rulings with short summaries to study the performance of text summarization systems. Wrzalik and Krechel [18] release a dataset for legal information retrieval (IR), which is based on case documents from the Open Legal Data platform [9]. Chalkidis et al. [19]

[9] https://eur-lex.europa.eu/

present FairLex, a multilingual fairness benchmark of four legal datasets that covers five languages and five sensitive attributes. They employ FairLex to evaluate the fairness of pre-trained language models (PLMs) and the techniques used to fine-tune them. Holzenberger and Durme [5] introduce the SARA dataset to investigate the performance of natural language understanding approaches on statutory reasoning Waltl et al. [20] present a automated classification of legal norms with regard to their semantic type and propose a semantic type taxonomy for norms in the German civil law domain.

5.2. NLP Approaches in Legal Domain

Dozier et al. [21] discusses NER and named entity disambiguation (NED) in legal documents such as US case law, depositions, pleadings, etc. Glaser et al. [22] evaluate NER and NED approaches on a manually annotated German court decisions dataset. Chalkidis et al. [23] apply sequence labeling techniques to extracting core information from contracts. Large PLMs are usually trained using generic corpora and tend to underperform in specialized domains [24, 25]. Chalkidis et al. [2] apply BERT models [26] to English downstream legal tasks: text classification and sequence labeling, by exploring different pretraining and fine-tuning strategies.

Andrew [27] uses statistical and rule-based techniques to extract entities such as names, organizations and roles and their relations in legal documents. Chen et al. [28] propose a legal triplet extraction system for drug-related criminal judgment documents. Hong et al. [29] perform IE of case factors from a dataset of parole hearings. Cardellino et al. [30] employ IE in legal texts to recognize mentions of entities and links them to a structured knowledge representation[10]. Lüdemann et al. [31] use KG's to model business entities of multinational companies and employ it for tax planning strategies.

6. Conclusion and Future Work

In this work, we investigated extracting relevant key figures from legislative texts. To this end, we provided a universally applicable annotation schema together with a semantic model for key figures and their properties in legal texts. We successfully applied the schema and the model to legal texts. Moreover, we presented a dataset manually annotated by tax experts, which includes 85 annotated paragraphs from 14 different German tax acts with 157 annotated tax key figures as well as a knowledge graph populated from these annotated paragraphs based on our semantic model.

We evaluated state-of-the-art entity extraction models on the proposed dataset, facing the challenges of the low-resource scenario and long entity spans. The results showed that all models perform well for classes with low complexity and sufficient training data available. Nonetheless, for more complex entities the transformer-based language models significantly outperform the other models. However, as a limitation, such models also require a certain amount of training data to achieve acceptable performance. We further provided a transformer-based relation extraction approach using typed entity markers, which has performed very well in our experiments. Moreover, we introduced *task triggers* for training a combined model for entity and relation extraction and for different groups of entities with overlapping annotations. We have shown that comparable performance can be achieved with this combined model as with separately trained models. Using a combined model saves computational resources for training and memory resources for inference.

We make our dataset together with the semantic model and the KG, as well as the implementation of the entity and relation extraction approaches investigated in this work publicly available[11]. To showcase our work, we also provide a simple demonstrator application[12].

Future Work In the future, we also plan to consider alternative modeling approaches of the entity and relation extraction task, e.g., as a span-based classification, using machine reading comprehension or unsupervised approaches utilizing large PLMs. Even with the relation extraction approach used in this work, a more comprehensive evaluation can be performed by considering different entity markers and providing more or less information about the entities, such as the entity types. The KG's populated from the extracted key figures allows as next step to compare the KG's of existing and new law texts in terms of their key figures. In this future work, we also plan to evaluate other approaches for differential analysis and then compare them to the semantic approach described in this work. These detected changes then provide the input for an application to predict the impact of the law change on the expected tax revenue. The ontology developed in this work on the basis of German tax acts can thereby also be applied universally to other legal fields and languages.

Acknowledgments

The authors acknowledge the financial support by the German Federal Ministry of Finance in the project *"KISS - KI-gestütztes System zur Steueranalyse"*.

[10] LKIF ontology: http://www.estrellaproject.org/lkif-core/

[11] https://github.com/danielsteinigen/nlp-legal-texts

[12] https://huggingface.co/spaces/danielsteinigen/NLP-Legal-Texts

References

[1] Namysł, Marcin, Robust Information Extraction From Unstructured Documents, Ph.D. thesis, Rheinische Friedrich-Wilhelms-Universität Bonn, 2023. URL: https://hdl.handle.net/20.500.11811/10560.

[2] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, I. Androutsopoulos, LEGAL-BERT: The muppets straight out of law school, in: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online, 2020, pp. 2898–2904. URL: https://aclanthology.org/2020.findings-emnlp.261. doi:10.18653/v1/2020.findings-emnlp.261.

[3] M. Palmirani, G. Governatori, A. Rotolo, S. Tabet, H. Boley, A. Paschke, Legalruleml: Xml-based rules and norms., RuleML America 7018 (2011) 298–312. doi:10.1007/978-3-642-24908-2_30.

[4] J. Moreno Schneider, G. Rehm, E. Montiel-Ponsoda, V. Rodríguez-Doncel, P. Martín-Chozas, M. Navas-Loro, M. Kaltenböck, A. Revenko, S. Karampatakis, C. Sageder, J. Gracia, F. Maganza, I. Kernerman, D. Lonke, Lynx: A knowledge-based ai service platform for content processing, enrichment and analysis for the legal domain, Information Systems 106 (2022) 101966. URL: https://www.sciencedirect.com/science/article/pii/S0306437921001563. doi:https://doi.org/10.1016/j.is.2021.101966.

[5] N. Holzenberger, B. V. Durme, Factoring statutory reasoning as language understanding challenges, in: C. Zong, F. Xia, W. Li, R. Navigli (Eds.), Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021, Association for Computational Linguistics, 2021, pp. 2742–2758. URL: https://doi.org/10.18653/v1/2021.acl-long.213. doi:10.18653/v1/2021.acl-long.213.

[6] J.-C. Klie, M. Bugert, B. Boullosa, R. E. de Castilho, I. Gurevych, The inception platform: Machine-assisted and knowledge-oriented interactive annotation, in: Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations, 2018, pp. 5–9.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[8] B. Chan, S. Schweter, T. Möller, German's next language model, in: Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 6788–6796. URL: https://aclanthology.org/2020.coling-main.598. doi:10.18653/v1/2020.coling-main.598.

[9] M. Ostendorff, T. Blume, S. Ostendorff, Towards an open platform for legal information, in: Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, JCDL '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 385–388. URL: https://doi.org/10.1145/3383583.3398616. doi:10.1145/3383583.3398616.

[10] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 8440–8451. URL: https://aclanthology.org/2020.acl-main.747. doi:10.18653/v1/2020.acl-main.747.

[11] I. Beltagy, M. E. Peters, A. Cohan, Longformer: The long-document transformer, 2020. URL: https://arxiv.org/abs/2004.05150. doi:10.48550/ARXIV.2004.05150.

[12] M. Sagen, Large-Context Question Answering with Cross-Lingual Transfer, Master's thesis, Uppsala University, Department of Information Technology, 2021.

[13] W. Zhou, M. Chen, An improved baseline for sentence-level relation extraction, in: Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), Association for Computational Linguistics, Online only, 2022, pp. 161–168. URL: https://aclanthology.org/2022.aacl-short.21.

[14] I. Chalkidis, I. Androutsopoulos, A. Michos, Extracting contract elements, in: Proceedings of the 16th Edition of the International Conference on Articial Intelligence and Law, ICAIL '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 19–28. URL: https://doi.org/10.1145/3086512.3086515. doi:10.1145/3086512.3086515.

[15] I. Chalkidis, E. Fergadiotis, P. Malakasiotis, I. Androutsopoulos, Large-scale multi-label text classification on EU legislation, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 6314–6322. URL: https://aclanthology.org/P19-1636. doi:10.18653/v1/P19-1636.

[16] E. Leitner, G. Rehm, J. Moreno-Schneider, A dataset of German legal documents for named entity recognition, in: Proceedings of the 12th Language Re-

sources and Evaluation Conference, European Language Resources Association, Marseille, France, 2020, pp. 4478–4485. URL: https://aclanthology.org/2020.lrec-1.551.

[17] I. Glaser, S. Moser, F. Matthes, Summarization of German court rulings, in: Proceedings of the Natural Legal Language Processing Workshop 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 180–189. URL: https://aclanthology.org/2021.nllp-1.19. doi:10.18653/v1/2021.nllp-1.19.

[18] M. Wrzalik, D. Krechel, GerDaLIR: A German dataset for legal information retrieval, in: Proceedings of the Natural Legal Language Processing Workshop 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 123–128. URL: https://aclanthology.org/2021.nllp-1.13. doi:10.18653/v1/2021.nllp-1.13.

[19] I. Chalkidis, T. Pasini, S. Zhang, L. Tomada, S. Schwemer, A. Søgaard, FairLex: A multilingual benchmark for evaluating fairness in legal text processing, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 4389–4406. URL: https://aclanthology.org/2022.acl-long.301. doi:10.18653/v1/2022.acl-long.301.

[20] B. Waltl, G. Bonczek, E. Scepankova, F. Matthes, Semantic types of legal norms in german laws: classification and analysis using local linear explanations, Artificial Intelligence and Law 27 (2019) 43–71. doi:10.1007/s10506-018-9228-y.

[21] C. Dozier, R. Kondadadi, M. Light, A. Vachher, S. Veeramachaneni, R. Wudali, Named Entity Recognition and Resolution in Legal Text, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 27–43. URL: https://doi.org/10.1007/978-3-642-12837-0_2. doi:10.1007/978-3-642-12837-0_2.

[22] I. Glaser, B. Waltl, F. Matthes, Named entity recognition, extraction, and linking in german legal contracts, in: IRIS: Internationales Rechtsinformatik Symposium, 2018, p. 325–334.

[23] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, I. Androutsopoulos, Neural contract element extraction revisited, in: Workshop on Document Intelligence at NeurIPS 2019, 2019. URL: https://openreview.net/forum?id=B1x6fa95UH.

[24] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (2019) 1234–1240. URL: https://doi.org/10.1093/bioinformatics/btz682. doi:10.1093/bioinformatics/btz682.

[25] I. Beltagy, K. Lo, A. Cohan, SciBERT: A pretrained language model for scientific text, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3615–3620. URL: https://aclanthology.org/D19-1371. doi:10.18653/v1/D19-1371.

[26] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.

[27] J. J. Andrew, Automatic extraction of entities and relation from legal documents, in: Proceedings of the Seventh Named Entities Workshop, Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 1–8. URL: https://aclanthology.org/W18-2401. doi:10.18653/v1/W18-2401.

[28] Y. Chen, Y. Sun, Z. Yang, H. Lin, Joint entity and relation extraction for legal documents with legal feature enhancement, in: Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 1561–1571. URL: https://aclanthology.org/2020.coling-main.137. doi:10.18653/v1/2020.coling-main.137.

[29] J. Hong, D. Chong, C. Manning, Learning from limited labels for long legal dialogue, in: Proceedings of the Natural Legal Language Processing Workshop 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 190–204. URL: https://aclanthology.org/2021.nllp-1.20. doi:10.18653/v1/2021.nllp-1.20.

[30] C. Cardellino, M. Teruel, L. A. Alemany, S. Villata, A low-cost, high-coverage legal named entity recognizer, classifier and linker, in: Proceedings of the 16th Edition of the International Conference on Articial Intelligence and Law, ICAIL '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 9–18. URL: https://doi.org/10.1145/3086512.3086514. doi:10.1145/3086512.3086514.

[31] N. Lüdemann, A. Shiba, N. Thymianis, N. Heist, C. Ludwig, H. Paulheim, A knowledge graph for assessing agressive tax planning strategies, in: J. Z. Pan, V. Tamma, C. d'Amato, K. Janowicz, B. Fu, A. Polleres, O. Seneviratne, L. Kagal (Eds.), The Semantic Web – ISWC 2020, Springer International Publishing, Cham, 2020, pp. 395–410.