Concept Hierarchy Extraction from Legal Literature

Sabine Wehnert
Otto von Guericke University
Magdeburg, Germany
sabine.wehnert@ovgu.de

David Broneske
Otto von Guericke University
Magdeburg, Germany
david.broneske@ovgu.de

Stefan Langer
Legal Horizon AG
Magdeburg, Germany
stefan.langer@legalhorizon.ag

Gunter Saake
Otto von Guericke University
Magdeburg, Germany
gunter.saake@ovgu.de

ABSTRACT

Due to the ever-increasing amount of legal regulations, it became an interest of scholars to find ways of capturing domain-relevant knowledge and facilitate the navigation in legal text corpora. Furthermore, the contextual nature of legislation requires enhanced semantic capabilities to identify relevant regulations for specific user needs. This work aims for collecting concept hierarchies from German literature in the legal domain which are then integrated into a knowledge base with multiple clusters, allowing for different perspectives and efficient lookups. Having references to regulations in the leaves of the concept tree and higher levels with an increasingly abstract context, the resulting hierarchies provide the basis for creating legal domain knowledge in German law. Starting with rule-based annotation, we cluster extracted references, given their context features derived from tables of contents and *reasons for citing* from various textbook formats. We study the expressiveness of the obtained reference context features. Since different authors have their own notion of hierarchy given by the table of contents, we propose a heterogeneous lightweight ontology allowing for the coexistence of similar, yet diverse concept hierarchies to dynamically determine the best fit for a user in a semi-supervised setting. This approach is novel, since state-of-the-art ontologies are conventionally modeled under full integration and in a top-down manner, often not accounting for perspectives in knowledge representation.

CCS CONCEPTS

• **Information systems → Document representation**; *Personalization*; *Recommender systems*; *Link and co-citation analysis*; • **Computing methodologies → Information extraction**; Instance-based learning; • **Applied computing →** *Annotation*;

KEYWORDS

Rule-based information extraction, Legal literature, Concept learning, Web knowledge, Lightweight ontology

ACM Reference Format:
Sabine Wehnert, David Broneske, Stefan Langer, and Gunter Saake. 2018. Concept Hierarchy Extraction from Legal Literature. In *Proceedings of CIKM 2018 Workshop on Legal Data Analytics and Mining (LeDAM 2018).* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Nowadays, enterprises as well as lawyers are facing the challenge of keeping track of an overwhelming number of legal texts from different jurisdictions. Yet, it is their obligation to ensure compliance, so that often manual efforts are made to monitor changes in law. On the other hand, this means that new developments need to be integrated into already existing knowledge, e.g., if a law is amended and impacts other regulations which are used in a specific scenario, the knowledge needs to be adapted accordingly. There is a need for context-sensitive search and a grouping method which ensures that all relevant documents are retrieved for a specific situation. The natural language processing (NLP) community has made many advances, such as building citation networks [34, 38]. Surprisingly, there are few works addressing the extraction of legal concept hierarchies based on implicit semantic relations between legal texts. We define implicit semantic relations as relationships among legal texts which only apply in specific contexts, so that they are not coded as explicit citations within generally applicable regulations. For example, depending on the expertise of a lawyer (i.e., knowledge about implicit semantic relations), he can use his background to identify connected laws which are important for a specific case.

In this paper, we propose a method to extract information from a large number of textbooks. It can be used to identify contextually relevant texts based on their mentions within literature, providing evidence of a semantic relationship between legal texts depending on their closeness within the resulting concept hierarchy. This form of domain knowledge is modeled in a bottom-up manner, using the references to legal texts in the literature as instances in the bottom levels of the concept hierarchy. Above, descriptive context representations are desired, which we refer to as *reasons for citing*, for each respective regulation. These representations and relationships can be modeled according to the desired expressiveness of the resulting ontology. Winkels et al. show that *reasons for citing* can be extracted from the sentence referring to the respective regulation, and narrow them down to four relationship categories: *selection*, *application*, *concluding (denying)* and a category for *in relation to* [37]. Zhang and Koppaka link relevant legal texts based on *reasons for citing* and let experts assess their contextual quality [38].

LeDAM 2018, October 2018, Turin, Italy
© 2018 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

There are works addressing legal text linking based on the information given therein [7, 15]. These approaches use explicit citations from within the document itself or its metadata. We choose to use external knowledge from literature to find relationships which cannot be directly detected within these documents. For this, we model relationships among legal texts in a concept hierarchy, founded upon the spatial co-occurrence of their mentions in legal literature.

Our approach is therefore a step in a new direction of legal informatics, because we consider legal literature as a source of concept hierarchies to build domain knowledge. We base our method on the assumption that a (sub-) chapter headline corresponds approximately to the concept described in the section. Furthermore, the cited legal texts in each passage are seen as semantically related to the discussed concept of the respective section. While this assumption does not always hold - especially in cases where authors use creative titles - our studied literature contains descriptive concepts in most headings of sections.

For the scope of this paper, we establish a connection between legal documents which co-occur in the same chapter, part, section or lower level subsections. By means of a concept hierarchy, we are able to identify closely related legal texts in the lower parts, as well as those which have a higher distance given only one common concept on a high abstraction level. A limitation of this approach is that we extract and maintain explicit keywords forming a concept. Hence, we do not integrate it into a common understanding of standardized concepts, as it can be encountered in standard ontologies. Having legal textbooks of many different formats and authors as data sources, we expect many contradictions to occur during an attempt to establish mapping rules for a standard axiomatic ontology. Therefore, we follow a different notion of knowledge representation.

Similar to the process of studying law, we aim for a diversity of perspectives within our system, which are chosen depending on the context. Specifically, we are interested in the effects of letting a concept hierarchy remain in its original structure, derived from the table of contents (TOC), and coexist among other similar concept hierarchies belonging to the same cluster. In this work, we show how such an approach can model the contextual application of regulations and how it is able to adapt to user-given feedback. Thus, the contribution of this work is a combination of the following techniques:

- We apply rules to annotate elements in a textbook.
- We access DBpedia knowledge for named entity resolution.
- We form concept hierarchies and evaluate their components.
- We group concept hierarchies with nominal clustering.
- We discuss the use of heterogeneous lightweight ontology clusters for legal texts.

The remainder is structured as follows: Section 2 contains related work regarding concept hierarchy extraction, lightweight ontologies and the formation of clusters. Since our approach is derived from observations of research gaps for our specific use case, we provide a justification of our methods alongside. In Section 3, we describe our method of extracting concept hierarchies from legal literature and the subsequent steps of constructing the domain knowledge. We discuss experimental results in Section 4. Finally, we conclude our findings and unveil future research potential.

2 RELATED WORK

We introduce three main aspects regarding our aim of capturing and applying knowledge from textbooks. The concept hierarchy is derived from the inherent structure of a piece of literature. In this section, we first name some alternative approaches to extract concept hierarchies. Second, we provide the background for the formation of our knowledge base, being derived from a heterogeneous ontology. Third, we briefly outline a clustering method because it provides some further optimization options to control the cluster formation of a heterogeneous ontology.

2.1 Concept Hierarchy Extraction

Concept hierarchies are a means for representing knowledge in a hierarchical manner, having nodes of increasing abstraction per level and things as instances in the leaves of the tree. We intend to represent links between legal texts by shared concepts: The higher a linking node between two instances is located in the concept hierarchy, the more distant are two documents. There are several approaches for extracting concept hierarchies from unstructured text. Among them, we find rules to detect hyponomy relations based on Hearst Patterns [18], for example to represent legal vocabularies. Also eigenvector decomposition is a method for identifying term taxonomies [5]. Those patterns, however, are not applicable for the use case of linking legal texts. Lexical hyponomies are not suitable for references modeled as instances of the concept hierarchy tree, since the subsumption relation is not based on the vocabulary, but semantic relatedness gained from textbooks. Kuo et al. [22] propose hierarchical clustering to build concept hierarchies, while also the extraction of noun groups is a valid approach [25].

We examine methods of noun group extraction combined with hierarchical clustering further, and propose a combination of them for concept hierarchy extraction from literature. This approach is based on the assumption that an author captures the topic of a section within its title. In the highest levels of abstraction within our concept hierarchy, we gather elements from the Tables of Contents (TOC) within literature. Finally, we obtain a coarse- to fine-grained clustering of regulations based on the understanding of the corresponding author, while we assume that the *reasons for citing* in particular are relevant features justifying the cluster membership of a regulation.

Similar to this work, Günel and Aşlıyan [17] describe how to extract concepts from tutoring material in $\TeX$ format using domain relevance, entropy and lexical cohesion as inclusion criteria. Wang et al. extract concept hierarchies from textbooks by the *TOC* and Wikipedia [36]. We also use the *TOC* to find local relatedness of regulations given the section title and Wikipedia for Named Entity Resolution. Robin et al. compare two approaches for legal concept hierarchy extraction: hierarchical clustering and the extraction of topical expressions composed of noun groups [25]. Bruckschen et al. populate a legal ontology based on Named Entity Recognition [9]. In a related field, an approach using syntactic positions, called Formal Concept Analysis, is suggested by Cimiano et al. to extract concept hierarchies [11]. Based on topic modeling, part-of-speech tags and tf-idf weighting, Anoop et al. [2] suggest an unsupervised method for concept hierarchy extraction. A possible drawback of statistical topic modeling methods is the instability of retrieved topics and their keywords if the process is repeated on the same data. Belford et al. propose a method relying on matrix factorization to increase the stability and accuracy of topic models [6].

In contrast to these implementations, we use a rule-based approach to extract information. Legal applications can benefit from the control over data quality that a system designer has while using rule-based approaches, without compromising on the amount of data. Despite some deviations from the pattern - where authors incorporate creative headings for didactic purposes - we find very few of these cases in our collection of legal literature. We show the results of our approach in Section 4.

2.2 Heterogeneous Lightweight Legal Ontology

Despite some variation in the style format among the pieces of literature, another major challenge arises from the obtained concept hierarchies themselves: Initially, we obtain standalone hierarchies from each book, and the difference among them is unknown. However, topical overlaps are possible for diversified literature, thus posing a challenge in integrating all concept hierarchies in a non-contradicting manner.

Instead, we capture the contextual character of legal texts. Following the notion of hierarchical ontology clusters proposed in [31], we develop the idea of allowing multiple concept hierarchies to coexist without integrating them. Conventionally, one common language and understanding is desired for system architectures whose components access the same domain knowledge. Despite these advantages, for our application such an ontology requires high maintenance efforts resulting from frequent insertions of further knowledge, either by automatically determining valid mappings or checking for logically matching candidates.

In the legal domain, a common requirement is to ensure that all relevant documents are retrieved, thus we optimize for a high recall. This is however challenging when working with natural language, for example when encountering its cases of ambiguity, near-synonyms and polysemy. We therefore argue that concepts in legal literature may differ even for equal topics, which is due to different perspectives of the authors and their own interpretation. However, any human regularly overcomes these inconsistencies and ambiguity by either choosing one concept for a narrow but consistent understanding, or by broadening the scope and encompassing multiple sources to avoid omissions of important items, while accessing the most appropriate fit based on a contextual decision criterion. This criterion can be derived from user-provided feedback, for example by marking a document as irrelevant. Then, the concept hierarchy will be selected which most likely captures the user need based on the recomputation of relevance.

Since our intended knowledge base is built in a bottom-up manner, this work is different from axiomatic ontologies. There are legal ontologies available such as ALLOT [4] or LKIF [19], which are able to encompass multiple legal data sources, however also requiring alignment of the respective classes. These ontologies are built upon a document standard called Akoma Ntoso [32] and offer many ways of standardized information modeling on the document level and beyond. For our specific use case, we identify two possibilities to achieve our goal: Either an expert maintains contextual information regarding specific applications of laws together in such a standardized ontology - for instance, by using the contextual ontology language C-OWL [8] - or there is a system for legal literature covering different scenarios, user categories and jurisdictions, ideally resulting in a complete collection of all regulations needed for a case. Several bottom-up lightweight ontologies for legislative terms and entities exist [1, 10]. Our knowledge representation differs from these works substantially in terms of the application scenario and extraction method. To the best of our knowledge, there is no approach for the same use case within the legal domain allowing for a fair comparison with our work.

2.3 Concept Hierarchy Clusters

Given a large collection of textbooks, we apply clustering to increase contextuality and to reduce the search space for finding the the most applicable concept hierarchy for a context. As a result, many references from different concept hierarchies are merged together. In order to stucture the cluster, the distance information given by a hierarchical clustering algorithm can be exploited. For user-centered applications, a semi-supervised clustering method has been proposed by Bade and Nürnberger [3]. They introduce *must-link-before* constraints for clustering algorithms which can be applied to hierarchical agglomerative clustering. Those constraints identify instances to be linked and those which shall remain separate. Different from other works, this method also implies the means to model the hierarchical order of instances without requiring to define the exact level difference. As a use case for an enforced hierarchy, consider a scenario where a distance between European and national law is desired. After including *must-link-before* constraints, instances from the specified category are located closer to the reference instance than those which are forced to link on a higher node of the concept tree. The algorithm we use in the scope of this work allows for *must-link* and *cannot-link* constraints by defining a relationship between two features [23]. Due to space limitations, we leave the examination of constraint effects for future work and implement the clustering algorithm without constraints.

3 CONCEPT EXTRACTION FOR HETEROGENEOUS ONTOLOGIES

Following relevant literature and the justification of our method, we outline our approach for building a heterogeneous ontology. In particular, we describe the process of annotating features in textbooks to obtain a contextual representation of the reference by means of concept hierarchy clusters. Figure 1 depicts the workflow.

(1) An electronic literature resource is converted into a `txt` file.
(2) The text is preprocessed by performing tokenization, sentence chunking, orthographic coreference resolution, parts-of-speech tagging, roman literal identification and named entity resolution using web knowledge from DBpedia.
(3) Rule-based annotation is applied to match *TOC* components (*Chapter*, *Part*, *Subchapter*, *Subsubchapter*), *CS* components (regulation name *REG*, DBpedia concept *DBp*, relationship *REL* and references *REF*.
(4) All annotations are extracted into a `csv` file, resulting in a table of tokens T with their respective annotation features.
(5) The file is treated as a lookup table and for each *TOC* component, boundaries are determined.

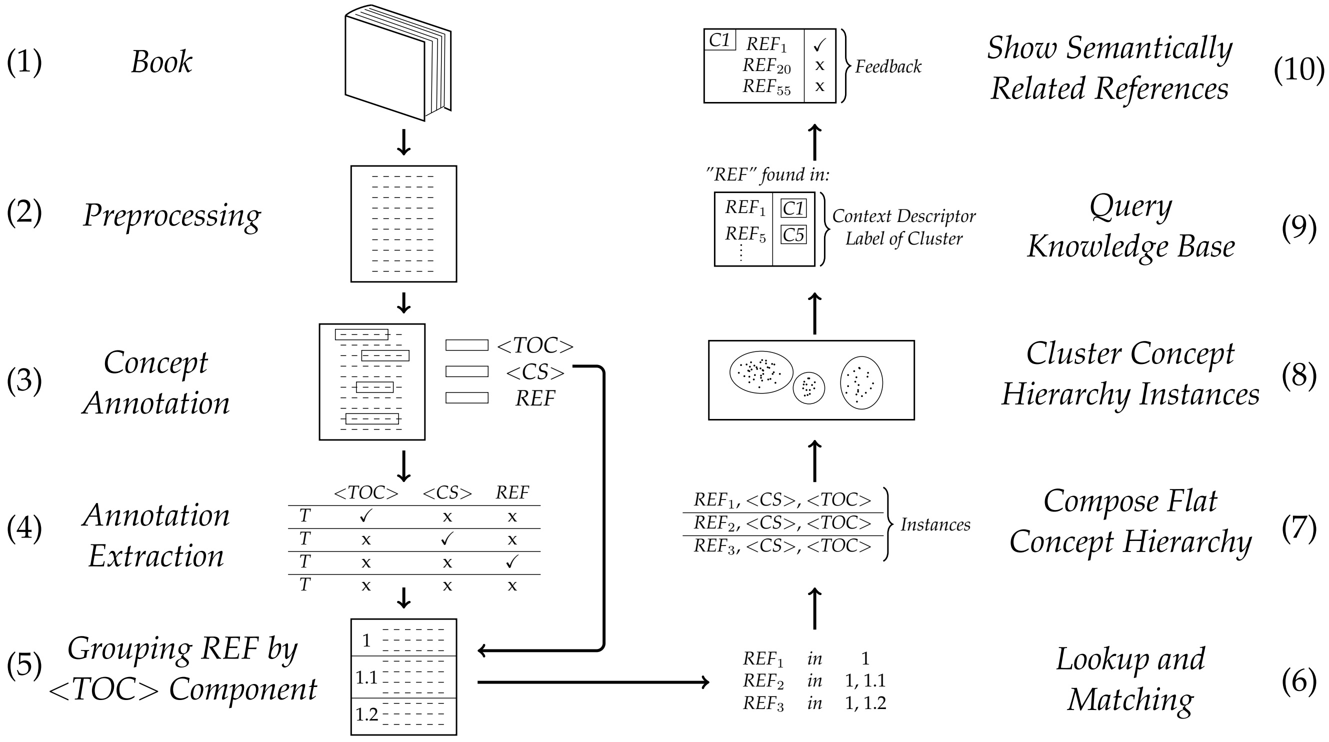


Figure 1: Workflow towards a lightweight heterogeneous ontology, used in a query expansion setting.

(6) All references are matched in document order to each *TOC* component with respect to the different section boundaries. Also, the *CS* information is retrieved from an extracted annotation file and assigned to the *REF*.

(7) After the feature information has been detected, a flat representation of the concept hierarchy is stored, with one *REF* instance per line and its *TOC* and *CS* feature information.

(8) The instances are clustered, using their nominal features.

(9) We included a possible use case, where a user searches for context information of a regulation REF_1. Here, for REF_1, cluster context descriptors *C1* and *C5* are retrieved. The user decides for *C1* and receives references linked to REF_1 depending on the data contained in the respective concept hierarchy cluster.

(10) A feedback mechanism can be implemented to narrow down relevant references. Different from our idea, Boonchom and Soonthornphisaj use term frequency-based ontology seeds for a legal ontology search task [29]. A similar approach for query expansion using a hierarchical legal knowledge base is by Schweighofer et al. [28]. Yet, their relevance feedback is based on the preferences of other users, unlike our approach focusing only on content.

Selected process steps to obtain the knowledge base are described in more detail in the following. We share more implementation details and program code on GitHub.[1]

3.1 Annotation

Since digital literature is conventionally available in PDF format, making use of formatting information similar to the approach of Günel and Aşlıyan on corresponding TeX-files can be cumbersome [17]. Alternatively, we convert the PDFs into `txt` files to speed up subsequent preprocessing steps. We use GATE - a widely adopted framework for text processing to preprocess the text - and JAPE Grammar rules [2] to annotate the concept hierarchy elements. For example, based on the pattern of a book publisher for a *TOC*, we specify matching criteria including orthographic information, roman numerals and part-of-speech tags [3]. The patterns for *reasons for citing* are described in Equation (1) and for the respective relationship in Equation (2).There is a trade-off between statistical and rule-based approaches: the former is faster to implement but less accurate, the latter is slow to implement but more accurate. Waltl et al. emphasize the effectiveness of rule-based information extraction due to explicitly applied domain knowledge and suggest this approach as an alternative to machine learning algorithms, since the latter often require a sufficient quality of training data [33]. Regarding the annotation of several elements within a textbook, we define rules suited for the respective elements which we consider as expressive features. We proceed with a description of these rules for TOCs, *reasons for citing* and regulations.

[1] `https://github.com/anybass/HONto`

[2] `https://gate.ac.uk/sale/tao/splitch8.html`

[3] We use the German model `german-hgc.tagger` from the Stanford parser `https://nlp.stanford.edu/software/tagger.shtml`

3.1.1 Table of Contents (TOC).

Depending on the publisher, a table of contents manifests itself in various styles. From numeral-only versions to mixed alphabet, roman literal and numeric variations, we define separate rules to capture each distinct heading element including its level in the context of the table of contents. Despite the efforts in rule definition, there are not many substantial variations within each publishing style, so that minor inconsistencies may be captured by generalization from seen examples. Waltl et al. combine the advantages of rule-based approaches with those of machine learning techniques because domain knowledge can be directly incorporated into the training phase to obtain more control over results [33]. However, it is out of scope of this work to train an annotation classifier and a potential future optimization task. After annotation, we export the *TOC* features. Based on the detected elements, we determine the boundaries for each level of the *TOC* hierarchy to store the respective references contained per part, subchapter and subsubchapter.

3.1.2 Reasons for Citing (RFC) and Relationships (REL).

Each sentence with a reference to a legal text potentially contains information about the rationale of this citation, which serves as a contextual summary. We divide the citation summary *CS* into the regulation name *REG*, the reason for citing *RFC* - following the notion of an entity - and its relationship *REL* with the regulation, captured by verb forms. Extracting the *CS* serves as feature information for a clustering algorithm. Another application is in connection with a reasoner based on the abstract relationships. Similar to the approach of Winkels et al. [37], a model of relationships among legal texts can be derived from textbooks and then be incorporated into the concept hierarchy. In addition, reasons for citing *RFC* can be considered for the user of a (content-based) legal recommender system as an explanatory component, to be displayed alongside the reference as a context descriptor. We find several pattern varieties proposed for keyphrase extraction and consider them for the *RFC* [21, 35]. While the respective authors analyze English language and capture adjective groups in addition to noun groups as well, there are more distinctions available for part-of-speech-tags in German language. Since including all adjective groups results in a larger number of distinct nominal features, we limit the pattern to minor sequence variations allowing for attributive adjectives. In our use case, we define the following expression to capture the *RFC*:

$$\begin{aligned} RFC = (&NN \mid NNS \mid NNP \mid NNPS \mid NE \mid \\ &(NN\,(ADJA \mid NN)*\ NN))+ \end{aligned} \tag{1}$$

Due to space limitations, this pattern is a simplified version of the actual one, here only listing candidate part-of-speech tags (POS) using the SSTS tagset [27]. Our rules account for a variety of possible sentence structures in German natural language. Those patterns which are formulated by using the more expressive JAPE rule syntax are defined with priorities, so that the most restrictive rule is applied first. Likewise, there are patterns for relationship extraction examined by multiple authors, as well [13]. We adapted them to German language and added negation tags with

$$REL = (PTKNEG \mid V\text{-}INF \mid V\text{-}PP \mid V\text{-}FIN)+ \tag{2}$$

as the simplified relationship pattern *REL*. In the verb categories we subsume the tags using a hyphen, for example *V-INF* is a placeholder for *VAINF*, *VVINF* and *VMINF*, which are originally output by the Stanford parser. The relationship feature of the annotation in this case is formed as a concatenation of *REL* matches within a sentence containing *RFC*. We adjust the matching rule regarding specific word patterns for important indicators - strings indicating contradictions (e.g., in German “Widerspruch") or selections (e.g., in German “Beispiel") - which cannot be generalized with parts-of-speech information. Also, if there is a syntactic indication of a legal term definition (e.g., in German “nach" or “gemäß") within a law, we fill undetected *REL* fields with an is-relationship (in German: “ist"). Furthermore, we clean the matches by parsing out non-descriptive strings for a relationship between a reference and its reason for citing (e.g., in German “denke"). This consequently results in sparse relationship features, since the above rules are both specified within sentence boundaries. While our assumption that a sentence citing a regulation contains *RFC* and *REL* patterns, this is not always the case. For the subsequent steps, we only consider those regulations containing *RFC*, and optionally *REL*. Any annotated regulation contained in the document where *RFC* is missing may not hold enough context information to determine its applicability for the context. Despite this limitation, it shall not have severe consequences in case of a sufficiently large heterogeneous ontology, since other extracted concept hierarchies for the same context shall cover possible gaps due to the highly regularized nature of legislation.

3.1.3 Regulations (REG, REF).

Many scholars have examined methods to extract regulations from unstructured text [34], often to create a citation network based on the references within the original regulation text [37]. While currently machine learning approaches remain popular, rule-based methods achieve high precision and recall, as well, which is due to the highly regularized pattern of regulation citation. In German law, there are fixed citation guidelines. Therefore, a sufficiently high proportion of citations can be detected with rules, with precision and recall in the range from 80% to 90% [34]. In addition, legal language contains term definitions, which are implicitly referenced by other laws [34]. Those term definitions can be extracted with rules and stored in a Lookup dictionary. Although it is out of scope of this work, we plan to analyze and enrich regulations with legal term definitions - to be found in other regulations - to gain more context information from the knowledge provided in the data source itself. We considered corner cases in reference citations, thus aiming for an improvement of the already high regulation coverage. These corner cases include references containing more than two regulations from different sources, and occurrences of connection indicators, in German abbreviated as “i. V. m.". These annotations shall contribute to a rich knowledge base.

3.1.4 Access Web Knowledge (DBp).

Wang et al. suggest in their approach to apply web knowledge for identifying concept candidates [36]. We access Wikipedia-based linked open data through the DBpedia Spotlight [4] plugin for GATE [5]. Unlike their method, we intend the knowledge base to perform named entity resolution directly on the citation summary. If a DBpedia entry exists in the

[4] https://www.dbpedia-spotlight.org/

[5] GATE LODtagger component: http://www.semanticsoftware.info/lodtagger

sentence containing a reference, we split the URI to obtain the concept name as a nominal feature. We observe that most matches occur for the regulation or the *RFC* tokens. There is one frequent misclassification regarding the *German Civil Code* (BGB), where the DBpedia lookup yields a swiss political party instead of the civil code, which we manually corrected before composing the concept hierarchy. After having annotated the nine feature types (*Chapter*, *Part*, *Subchapter*, *Subsubchapter*, *REG*, *DBp*, *RFC*, *REL*, *REF*), we export them from GATE and build the concept hierarchy.

3.2 Compose Concept Hierarchies

Figure 2 shows how we compose and evaluate the concept hierarchy. In this example, there are two simplified concept hierarchies, which are obtained from the JAPE rule-based annotations. In the fictive *CS* node, we summarize the features *REG*, *DBp*, *RFC*, *REL* for space reasons, however, they are all stand-alone features. Each element has mandatory values for the *Chapter*, *RFC* and *Reference*. The other fields are optional because we do not assert that the rules return values for each feature.

Given the illustrated concept hierarchy in Figure 2, we evaluate the results by setting the *Chapter* as a class label - thus expecting a reproduction of the structure of a chapter - and by not including it in the features to be processed. As indicated by the arrows, the test data can match the learned examples by comparison of the subfeatures and early merges are an indicator for higher similarity between two instances. A possible limitation of this approach comes from the reliance on explicitly stated information. For instance, if the *RFC* are not indicated within the reference sentence or if they are faulty extracted, this can decrease the expressiveness of the features for the desired structure. Since the resulting concept hierarchy depends on the author of the book, his perspective may not be suitable for any user. Therefore, we see a possible remedy in the notion of concept hierarchy clusters, forming a heterogeneous lightweight ontology.

3.2.1 Concept Hierarchy Clusters. Extracting a narrow concept hierarchy with only nominal features leads to a lower probability of getting all relevant references for a specific information need. Consider the following example: While one book may focus on the aspects of national law, another depicts European legislation. In reality, this information needs to be considered as a whole, since European legislation supersedes national law.

Recalling the discussion from Section 2.2, we show how exactly a heterogeneous ontology can serve a user who is interested in complete, reliable and founded information. Aside from our experiment of matching extracted instances with *Chapter* labels, an actual application of this method is to classify for *Relevance* instead. Figure 3 illustrates how a heterogeneous ontology in legal contexts may emerge. In the setting of a recommender system, suppose there is a cluster containing two concept hierarchies with sets of instances (1, 5, 8) and (1, 2, 4, 8) respectively. In the first scenario depicted on the left hand side, the recommender system receives positive user feedback regarding instance 1. Since this instance is present in the current context which is more narrow than other concept hierarchy, the context is not altered. In contrast, a similarity function *(*A) receives negative feedback for instance 5 in the second scenario, thus resulting in a context switch to the other concept hierarchy without instance 5. There are several approaches for similarity adaptation, as investigated by Stober and Nürnberger in [30]. In addition, the heterogeneous ontology can also be used for query expansion, as previously pointed out regarding Figure 1.

We find that for a legal recommender system, heterogeneous ontologies - as defined in this work as clusters of concept hierarchies acquired from suitable literature - can indeed fulfill the following desirable functions:

(1) They group semantically related concept hierarchies.
(2) Their clusters allow for efficient lookups, instead of querying the whole ontology.
(3) They are sensitive towards user feedback.
(4) They are as relevant as possible by applying the narrowest context given user feedback constraints.

We conducted some experiments with subsets from the 78 documents (subchapters from three fixed chapters), the results are shown in the next Section 4.

4 RESULTS

To show the effect of adding knowledge to the heterogeneous lightweight ontology, we evaluate the annotation and perform two experiments. The first experiment applies COBWEB clustering on the features, without knowing the *Chapter* class label. The second approach is a classifier for the same features, this time we use the COBWEB tree. Before we present their results, we describe the experiment setting and evaluation measures.

4.1 Evaluation Setup

The aim of this evaluation is to determine the expressiveness of our selected features to distinguish between abstract concepts. In this work, we intend to show the feasibility of our proposed knowledge extraction and representation method. Therefore, we create clusters of semantically similar concept hierarchies by using the COBWEB algorithm [14]. It is a recursive hierarchical tree algorithm, which learns incrementally from new instances, given four options of incorporating them (creating a new child node, adding to an existing child node, merging two similar child nodes and incorporating the newest instance therein, and splitting a node, so that it becomes a child of the current node) [23]. We visualize our results by using the python library `concept_formation` [6] by MacLellan et al. [23]. Instead of incorporating several books, we evaluate this method with respect to the most high-level concepts (i.e., chapter titles) of one comprehensive book. In particular, we used chapters (1), (4) and (8) from Derleder et al. because they were perceived as topically related, while still treating different concepts [12]. For a rich heterogeneous ontology, multiple books need to be taken into account, among which several topical overlaps shall occur to compensate for losses from the extraction process or a different focus of an author. In case of significant overlaps, two concept hierarchies shall be merged.

4.2 Evaluation Measures

Regarding the annotation success, we determine the effectiveness of context feature extraction by computing the average coverage of

[6] `https://github.com/cmaclell/concept_formation`

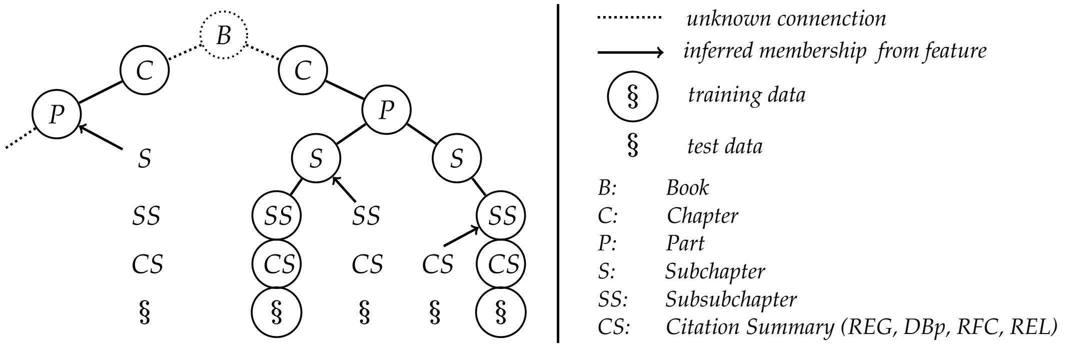


Figure 2: Structure and Evaluation of a Concept Hierarchy

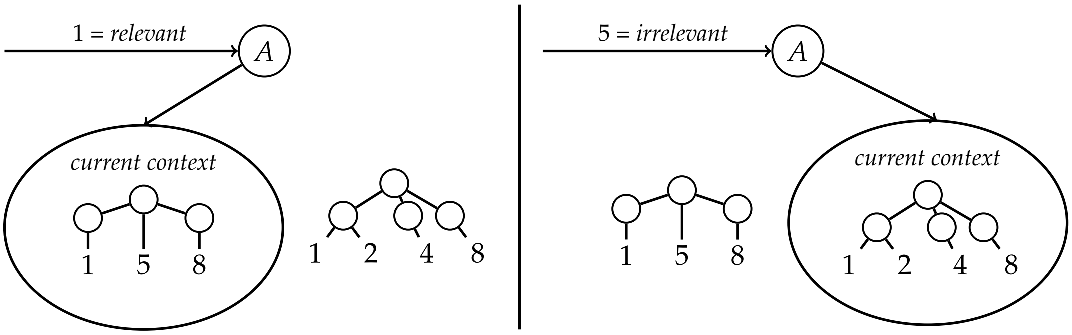


Figure 3: Incorporating user feedback in a cluster of concept hierarchies with an adaptation function (A)

references *REF* by *RFC* annotations. Basically, if a sentence contains a pattern which can be detected by our JAPE rules, there will be an *RFC* annotation. Since we only considered those regulations whose context features (especially *RFC*) could be retrieved, this evaluation is important to understand how many data points were the basis for the subsequent steps of clustering and classification.

Our evaluation measure for the supervised clustering experiment is the Adjusted Rand Index (ARI), originally proposed by Hubert and Arabie [20]. It quantifies the overlap between two partitioning approaches, in our case, we compare the COBWEB clustering and the class labels (i.e., textbook chapters). Its expected value 0 indicates a random clustering, while a value close to 1 corresponds to a high agreement between the resulting clustering and class label partitions. Santos and Embrechts suggest using the ARI for supervised multilabel classification evaluation due to its ability to measure the relationship of two elements instead of the correct class label assignment [26]. While we only use one book, we expect an ARI above 0.5 because each chapter contains unique themes and possible overlaps in cited regulations *REG*. Having heterogeneous ontology clusters, an automatic merging criterion can be applied to achieve clusters of topically related concept hierarchies. Based on the ARI, this merging criterion has been implemented by Pavan et al. to extend k-means clustering [24].

For the classification task, we use average values of precision and recall. Calculating average recall is rather unconventional [16], however, optimizing for a high recall is crucial in the legal domain. Those two measures quantify how well the COBWEB tree is be able to infer the correct class membership given the instance features, as shown in Figure 2. In particular, our average precision measures the percentage of correctly identified class members compared to all instances labeled as class members by the algorithm, averaged over the number of runs and all classes. The average recall in our case is defined as the fraction of correctly identified instances of a class compared to all that belong to the respective class, averaged over all runs and classes. Intuitively, a false positive recommendation of a regulation is not as severe as a false negative for the legal domain.

4.3 Evaluation of Annotation

We evaluate our annotation results regarding the number of detected references *REF* compared to the number of extracted *RFC* in the chapter, since we require the latter for concept formation. Spiegel-Rosing found for scientific texts descriptive *RFC* context in 80% of the sentences. We assume that in a German legal textbook, slightly less *RFC* will be detected, due to a different writing style (e.g., more complex syntax and longer sentences). Consequently, our aim for *RFC* annotation is set to 70% of *REF* occurrences. Therefore, we define a JAPE rule and annotate the text based on a pattern that is able to detect several citation formats:

German law: § *676 a Abs. 1 Satz 1 BGB*
German law: *Art. 1 und 2 Abs. 1 GG*
European law: *2000 / 46 / EG*

In Table 1, we list the number of reference annotations corresponding to the book chapters: (1) Bankvertragliche Grundlagen (English: Foundations of Banking Contracts), (4) Kapitalmarkt- und Auslandsgeschäfte (English: Capital Market and Foreign Transactions), (8) Europäisches Bankenrecht mit Länderabschnitten (English: European Banking Law by Country). Additionally, we indicate the number of *RFC* and the average percentage of detected *RFC* from all *REF* annotations per chapter. The numbers in the column header depict the document number, corresponding to the subchapters of the textbook. We find that almost 75% of the references have an annotation value for *RFC*. The restrictions we included in our pattern prevent us from extracting the chapter name as a *REF*, and despite some missing references and *RFC* due to long-range dependencies within the sentence or unwanted headline text insertions at page breaks, the noise in the text data (e.g., citations of other books in a reference-like format) did not affect the extraction substantially. Nevertheless, all subsequent steps depend on the annotation, so that a loss in this step propagates forward to the clustering and classification task.

4.4 Evaluation of Heterogeneous Legal Ontology

We evaluate our results for the COBWEB clustering algorithm using the extracted *Chapter* feature as the ground truth class. With the remaining context information starting with the *Part* feature until the *REF* feature, the instances are supposed to be grouped by the COBWEB clustering algorithm. In order to show the effect of a successful extraction method, we restricted the instances only to those cases where a value could be retrieved for the *Part* feature, since this is the most abstract class. To have an equal class distribution, we downsampled the instances of other chapters to match the class with the fewest instances left. This has not been achieved with a random selection, but instead we selected a group of instances which were previously spatially close in the textbook. This has the advantage of not missing important context, as well as limiting the variance in nominal features. For a fair comparison, running the evaluation with different instance groups yielded mostly similar results, however we observe that more variability leads to less similar examples and thus a lower ARI score.

For the first evaluation shown in Figure 4 with 2 principal components *p*, 3 Chapters and 1020 instances *i* of balanced classes, we obtain an adjusted rand index *(*ARI) of 0.28. Each axis holds one principal component analysis (PCA) dimension to visualize a projection of the cluster shape. According with our expectation, there are three clusters, while each cluster consists of two to three ellipsis shapes. The chapter labels in Figure 4 indicate that the algorithm does not have enough information to distinguish between chapter (1) (labeled as B) and chapter (4) (labeled as K) and chapter (8) (labeled as E). Many instances of particularly chapters (4) and (8) are placed in the wrong cluster. From this, we conclude that despite having balanced classes, there may be topical overlaps among the concept hierarchies which shall either result in a merge or are lacking evidence for separate groups. If we allow for a slight class imbalance of the instances by increasing the number of chapter (1) and (4) instances in a comparable amount to 1149, the ARI increases to 0.64, as shown in Figure 5. This also led to a different cluster shape and a better discrimination between the three chapter classes. The improvement can be seen in the classes, where more labels correspond to the cluster membership. It indicates that the clustering approach found more agreement between clusters and the ground truth classes. That observation lets us conclude that additional examples can lead to a higher ARI if they only broaden the feature value space moderately. In previous experiments, we applied the algorithm to all extracted instances, leading to an ARI of 0.05, presumably because of the high variance of instances within a chapter and different chapter length. Since this class imbalance will naturally occur in a heterogeneous ontology, we need to investigate futher how the approach scales and what the limitations are regarding the feature diversity.

We perform a second experiment on the same data, but in the classification setting with a COBWEB tree with 10 runs *r* and 300 training instances *num*. The result of the classification algorithm is

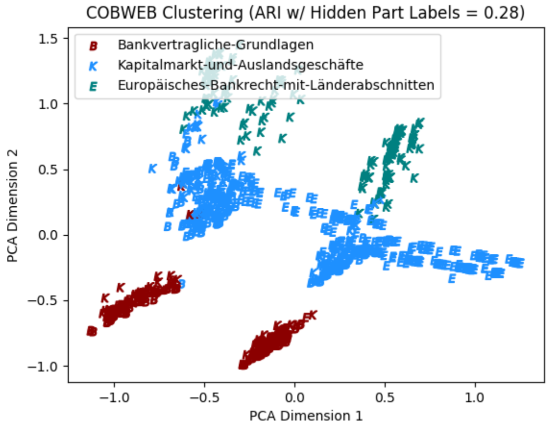


Figure 4: COBWEB clustering with p=2, i=1020 and the ARI evaluation [20]

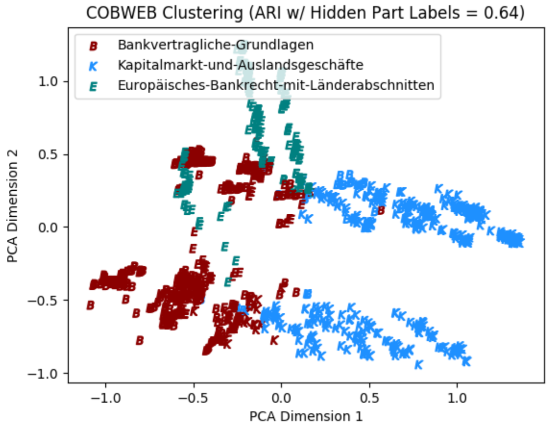


Figure 5: COBWEB clustering with p=2, i=1149 and the ARI evaluation [20]

Table 1: Evaluation of *REF* and *RFC* detection. From each chapter, we analyzed all subchapters.

(1)	1	2	3	4	5	6	7	8	9						**Avg. %**
REF	197	40	196	47	41	107	568	131	250						
RFC	170	30	168	37	31	83	385	74	160						**72**
(4)	50	51	52	53	54	55	56	57	58	59	60	61	62	63	**Avg. %**
REF	211	82	1091	283	119	41	82	283	270	483	112	115	164	237	
RFC	158	60	643	232	85	33	70	215	227	400	85	93	111	221	**74**
(8)	72	73	74	75	76	77	78								**Avg. %**
REF	47	90	188	40	67	28	370								
RFC	36	61	147	30	43	16	275								**73**

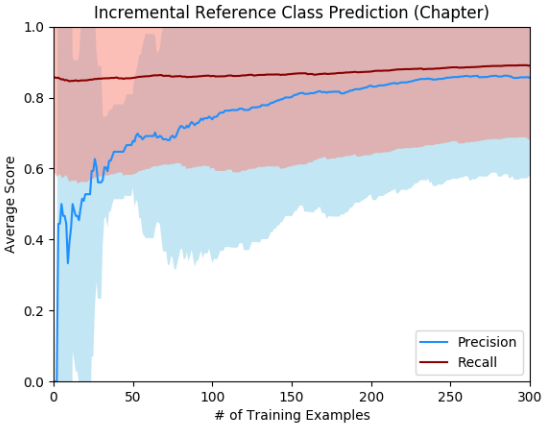


Figure 6: COBWEB tree with r=10, num=100, i=1020

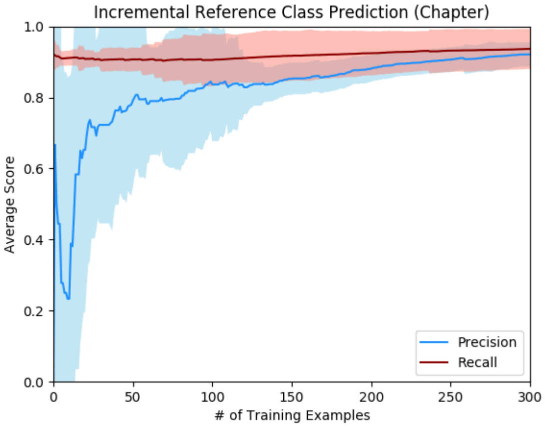


Figure 7: COBWEB tree with r=10, num=100, i=1149

shown in Figures 6 and 7, including 95% confidence intervals for the average precision and recall values. In Figure 6, the confidence intervals obtain a range of 40 percentage points (pp), witnessing of an unstable classification result of 80% precision and 87% recall on average after 200 training examples. The effect of adding further examples is illustrated in Figure 7 and similar to the previous experiment, which manifests in a gain in precision of about 10pp and a slight increase of 5pp in the average recall score. Please note that the range of the confidence interval is reduced to 20pp for recall and to 10pp for precision, which is a significant improvement of the classifier performance. In summary, the results for the COBWEB algorithm vary depending on the number of examples for each concept hierarchy. A recall of more than 90% is desirable, so that the results from the second setup of each experiment are regarded as sufficient evidence for descriptive features to distinguish between different contexts. We discuss the general applicability of the results.

4.5 Discussion

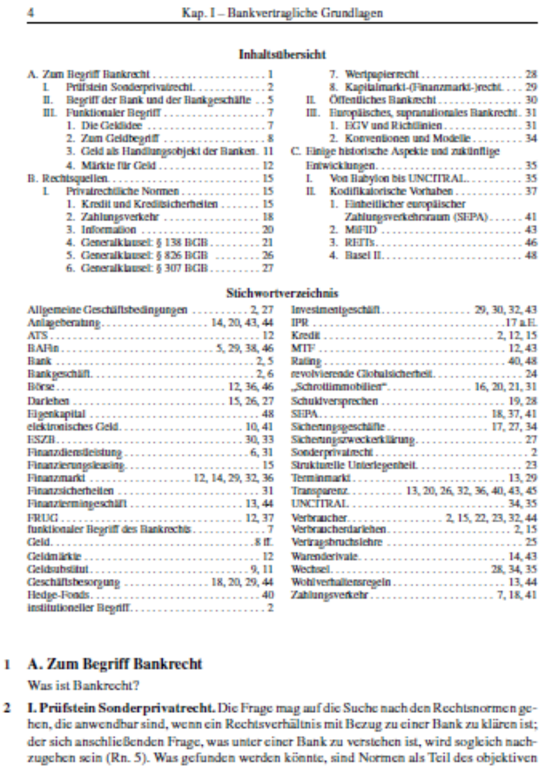
There is more research potential in the question whether this approach also works for other domain literature, or what happens if other clustering algorithms with advanced capabilities of constraint formulation are chosen. Considering that we used concept hierarchies mostly about general banking law, financial markets and european banking law, the overlap of *REG* and *RFC* is considerable. After other books about different subjects are added, those three concept hierarchies may form a cluster. During the concept hierarchy extraction, we found that there are four major limitations of our approach: First, literature resources are needed which cover the information need. Otherwise, a user may not find his case represented. Second, for each textbook, there can be a different format of citations or the *TOC* components. This results in a higher manual effort for rule formulation. Third, since we only had the PDF files of literature available, there were challenges in segmenting the file and assigning references to each section, leading to missing feature values. Fourth, despite having gained much domain information from the textbook, we need to investigate more methods of leveraging those. Since we plan to implement a lightweight heterogeneous ontology, we uncover future research fields in Section 5.

5 CONCLUSION AND FUTURE WORK

To conclude, our lightweight heterogeneous ontology is composed of concept hierarchies which are derived from literature. It is a promising area for further work. We pointed out the reasons for accepting coexisting perspectives in the legal domain and gave indications of how to take advantage of many sources, while still controlling the results with constraints and user feedback. The rule-based annotation method provided features for context-aware classification and clustering of the concept hierarchies. Overall, the results indicate that the chosen features, the extraction method and the `concept_formation` library are suitable for detecting semantic similarity in the book we selected. Regarding future work, we are curious about how this method performs, if additional features of the content of referenced regulations and term definitions are taken into account. Another field to study is the impact of abstract relationship categories on clustering. We see possible applications of the learned ontology in the field of law clustering, legal context search, topic detection and legal recommender systems and intend to explore more about these use cases.

A AN EXAMPLE OF THE WORKFLOW

We clarify the workflow described in Figure 1 by using the first subchapter of the book [12]: “§ 1 Grundlagen (Begriff, Geschichte, Rechtsquellen)".



4 Kap. I – Bankvertragliche Grundlagen

Inhaltsübersicht

Stichwortverzeichnis

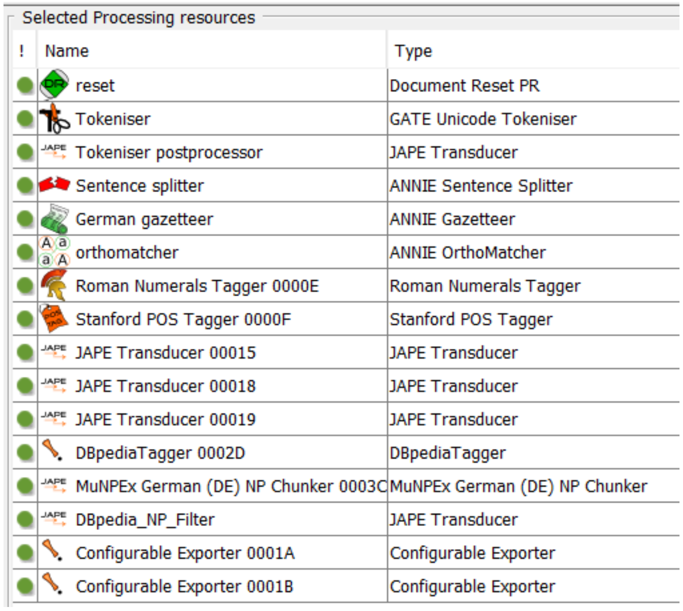
1 **A. Zum Begriff Bankrecht**

Was ist Bankrecht?

2 **I. Prüfstein Sonderprivatrecht.** Die Frage mag auf die Suche nach den Rechtsnormen gehen, die anwendbar sind, wenn ein Rechtsverhältnis mit Bezug zu einer Bank zu klären ist; der sich anschließenden Frage, was unter einer Bank zu verstehen ist, wird sogleich nachzugehen sein (Rn. 5). Was gefunden werden könnte, sind Normen als Teil des objektiven

Figure 8: Layout of the PDF source document

Starting with the subchapter as a PDF file which is shown in Figure 8, we convert it into a `txt` file using `pdftotext`[7]. This is due to the double-columnar format of the table of contents, which can be processed more accurately when applying the `raw` flag. The resulting text file is then imported into GATE.



Selected Processing resources

!	Name	Type
●	reset	Document Reset PR
●	Tokeniser	GATE Unicode Tokeniser
●	Tokeniser postprocessor	JAPE Transducer
●	Sentence splitter	ANNIE Sentence Splitter
●	German gazetteer	ANNIE Gazetteer
●	orthomatcher	ANNIE OrthoMatcher
●	Roman Numerals Tagger 0000E	Roman Numerals Tagger
●	Stanford POS Tagger 0000F	Stanford POS Tagger
●	JAPE Transducer 00015	JAPE Transducer
●	JAPE Transducer 00018	JAPE Transducer
●	JAPE Transducer 00019	JAPE Transducer
●	DBpediaTagger 0002D	DBpediaTagger
●	MuNPEx German (DE) NP Chunker 0003C	MuNPEx German (DE) NP Chunker
●	DBpedia_NP_Filter	JAPE Transducer
●	Configurable Exporter 0001A	Configurable Exporter
●	Configurable Exporter 0001B	Configurable Exporter

Figure 9: Overview of applied GATE processing resources

In Figure 9, we list the required processing resources. Therefore, the following GATE plugins[8] are needed:

- Lang_German
- Tagger_Numbers (Roman Numerals Tagger)
- Stanford_CoreNLP (model: `german-hgc.tagger`)
- DBpediaTagger
- Tools (Configurable Exporter)

The resources named *reset*, *Tokenizer*, *Tokenizer postprocessor*, *Sentence splitter*, *German gazetteer* and *orthomatcher* are part of the German IE System and used for preprocessing. Afterwards, the *Roman Numerals Tagger* adds roman numeral properties to numbers within the text, which is needed for the specific structure of the table of contents in the textbook. The *Stanford POS Tagger* adds part-of-speech tags to the tokens. Then, our own JAPE files are accessed and the annotation logic for *TOC, REF, RFC, REL* and the remaining features is applied. It is worth noting that the order of the rule execution has to be preserved. The reasons for citing *RFC* detection depends on previously identified references to legal documents *REF* within the text. Similarly, the relationships *REL* can only be found if *RFC* annotations have already been made. As a next step, the *DBpediaTagger* searches for any match within the text with a DBpedia concept. We only extract the concepts which are present in a sentence containing a reference. Figure 10 illustrates how each element of the *TOC* is captured by the JAPE rules.

On the other hand, Figure 11 shows how the elements of the citation summary have been detected within GATE. There are two *Configurable Exporters*. One of them extracts *TOC* information for each Token as shown in step 4 of Figure 1. The second one extracts

[7] `https://www.xpdfreader.com/pdftotext-man.html`

[8] `https://gate.ac.uk/gate/doc/plugins.html`

the sentences containing a reference *REF* with a corresponding reason for citing *RFC* and relationship *REL* components. The output of this process are two text files in csv-like format:

`1_Grundlagen_toc.txt`

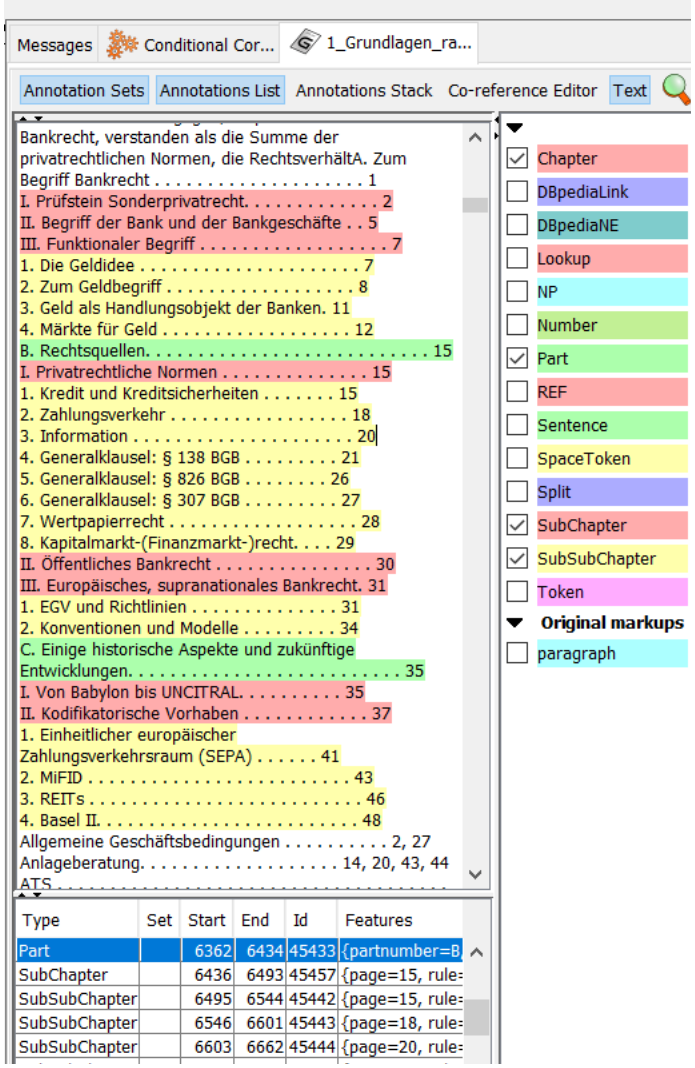
`1_Grundlagen_rel.txt`

Then, the python script `gate_postprocess.py` is executed. It performs the grouping and matching between the extracted annotations and the raw text file from the first step. Using regular expressions, each *TOC* component is searched within the text to divide it into chunks. This is a critical step because it assumes consistency between the *TOC* section headings and the section headings present in the text body.

A PDF file offers no spatial information regarding its content, making this grouping and subsequent reference assignment necessary. A file named

`1_Grundlagen_features.txt`

is created, which contains in each row all the successfully matched features of a reference, see step 7 of Figure 1. We refer to this representation as the flat concept hierarchy because it contains feature information to construct a concept hierarchy out of the instances. Since the most abstract *Part* values encompass many instances and therefore carry information of how to merge directly below the root level of the concept hierarchy, we consider this important feature mandatory and exclude cases where the *Part* could not be retrieved. Such constraints can contribute to the correctness of the subsequent cluster formation step.

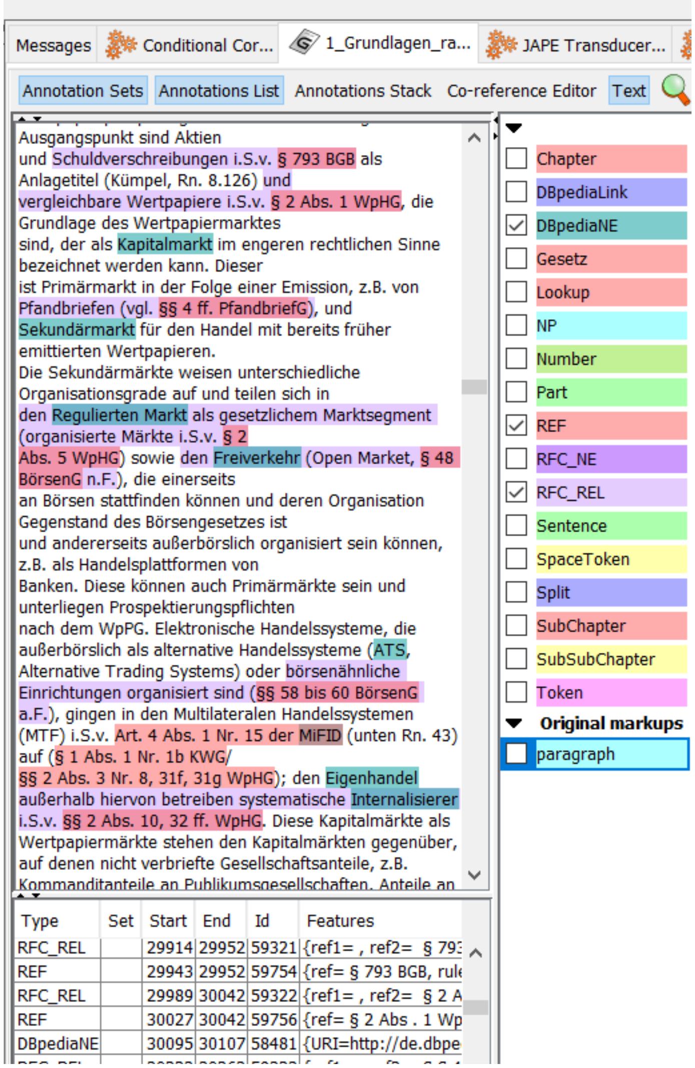


Bankrecht, verstanden als die Summe der
privatrechtlichen Normen, die RechtsverhältA. Zum
Begriff Bankrecht 1
I. Prüfstein Sonderprivatrecht. 2
II. Begriff der Bank und der Bankgeschäfte . . 5
III. Funktionaler Begriff 7
1. Die Geldidee 7
2. Zum Geldbegriff 8
3. Geld als Handlungsobjekt der Banken. 11
4. Märkte für Geld 12
B. Rechtsquellen. 15
I. Privatrechtliche Normen 15
1. Kredit und Kreditsicherheiten 15
2. Zahlungsverkehr 18
3. Information 20
4. Generalklausel: § 138 BGB 21
5. Generalklausel: § 826 BGB 26
6. Generalklausel: § 307 BGB 27
7. Wertpapierrecht 28
8. Kapitalmarkt-(Finanzmarkt-)recht. . . . 29
II. Öffentliches Bankrecht 30
III. Europäisches, supranationales Bankrecht. 31
1. EGV und Richtlinien 31
2. Konventionen und Modelle 34
C. Einige historische Aspekte und zukünftige
Entwicklungen. 35
I. Von Babylon bis UNCITRAL. 35
II. Kodifikatorische Vorhaben 37
1. Einheitlicher europäischer
Zahlungsverkehrsraum (SEPA) 41
2. MiFID 43
3. REITs 46
4. Basel II. 48
Allgemeine Geschäftsbedingungen 2, 27
Anlageberatung. 14, 20, 43, 44
ATS . . . .

☑ Chapter
□ DBpediaLink
□ DBpediaNE
□ Lookup
□ NP
□ Number
☑ Part
□ REF
□ Sentence
□ SpaceToken
□ Split
☑ SubChapter
☑ SubSubChapter
□ Token
Original markups
□ paragraph

Type	Set	Start	End	Id	Features
Part		6362	6434	45433	{partnumber=B
SubChapter		6436	6493	45457	{page=15, rule=
SubSubChapter		6495	6544	45442	{page=15, rule=
SubSubChapter		6546	6601	45443	{page=18, rule=
SubSubChapter		6603	6662	45444	{page=20, rule=

Figure 10: Annotated table of contents *TOC* components.



Ausgangspunkt sind Aktien
und Schuldverschreibungen i.S.v. § 793 BGB als
Anlagetitel (Kümpel, Rn. 8.126) und
vergleichbare Wertpapiere i.S.v. § 2 Abs. 1 WpHG, die
Grundlage des Wertpapiermarktes
sind, der als Kapitalmarkt im engeren rechtlichen Sinne
bezeichnet werden kann. Dieser
ist Primärmarkt in der Folge einer Emission, z.B. von
Pfandbriefen (vgl. §§ 4 ff. PfandbriefG), und
Sekundärmarkt für den Handel mit bereits früher
emittierten Wertpapieren.
Die Sekundärmärkte weisen unterschiedliche
Organisationsgrade auf und teilen sich in
den Regulierten Markt als gesetzlichem Marktsegment
(organisierte Märkte i.S.v. § 2
Abs. 5 WpHG) sowie den Freiverkehr (Open Market, § 48
BörsenG n.F.), die einerseits
an Börsen stattfinden können und deren Organisation
Gegenstand des Börsengesetzes ist
und andererseits außerbörslich organisiert sein können,
z.B. als Handelsplattformen von
Banken. Diese können auch Primärmärkte sein und
unterliegen Prospektierungspflichten
nach dem WpPG. Elektronische Handelssysteme, die
außerbörslich als alternative Handelssysteme (ATS,
Alternative Trading Systems) oder börsenähnliche
Einrichtungen organisiert sind (§§ 58 bis 60 BörsenG
a.F.), gingen in den Multilateralen Handelssystemen
(MTF) i.S.v. Art. 4 Abs. 1 Nr. 15 der MiFID (unten Rn. 43)
auf (§ 1 Abs. 1 Nr. 1b KWG/
§§ 2 Abs. 3 Nr. 8, 31f, 31g WpHG); den Eigenhandel
außerhalb hiervon betreiben systematische Internalisierer
i.S.v. §§ 2 Abs. 10, 32 ff. WpHG. Diese Kapitalmärkte als
Wertpapiermärkte stehen den Kapitalmärkten gegenüber,
auf denen nicht verbriefte Gesellschaftsanteile, z.B.
Kommanditanteile an Publikumsgesellschaften. Anteile an

□ Chapter
□ DBpediaLink
☑ DBpediaNE
□ Gesetz
□ Lookup
□ NP
□ Number
□ Part
☑ REF
□ RFC_NE
☑ RFC_REL
□ Sentence
□ SpaceToken
□ Split
□ SubChapter
□ SubSubChapter
□ Token
Original markups
□ paragraph

Type	Set	Start	End	Id	Features
RFC_REL		29914	29952	59321	{ref1= , ref2= § 793
REF		29943	29952	59754	{ref= § 793 BGB, rule
RFC_REL		29989	30042	59322	{ref1= , ref2= § 2 A
REF		30027	30042	59756	{ref= § 2 Abs . 1 Wp
DBpediaNE		30095	30107	58481	{URI=http://de.dbpe

Figure 11: Annotated section text with *REF*, *RFC*, *REL* and *DBp* components.

This flat concept hierarchy file can be used as a basis for a reasoning application, given the extracted hierarchy and yet to be defined axioms. For the selected subchapter, we visualized the tree structure which is shown in Figure 12.

The aim of this paper was to demonstrate the feasibility of the information extraction and to pave the path for future directions with an obtained knowledge representation. We studied the expressiveness of the features to obtain clusters (`concept_formation.py`) or to classify the instances (`concept_prediction.py`) correctly into their group given partial information. The results indicate that for each reference, the extracted features are sufficient for a correct context reconstruction. This is a desired property, given a large collection of textbooks where the context clusters shall contain related elements and all needed information for one context.

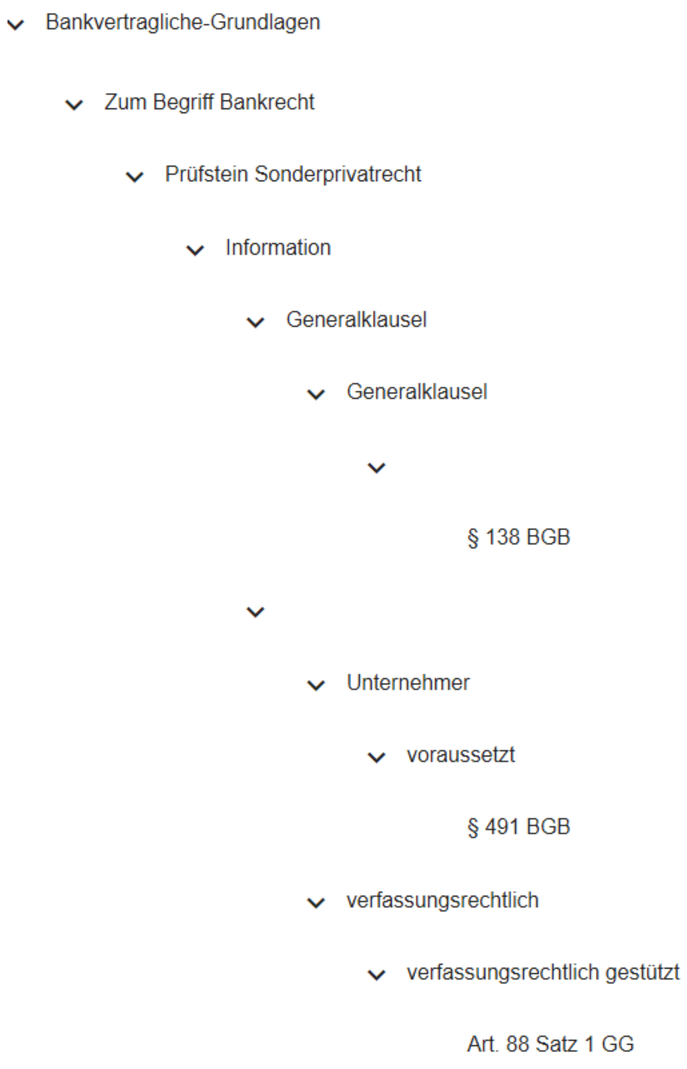


Figure 12: Excerpt of the extracted tree structure from the first subchapter of the textbook. The reference instances are located in the lowest level.

REFERENCES

[1] Gianmaria Ajani, Guido Boella, Luigi Di Caro, Livio Robaldo, Llio Humphreys, Sabrina Praduroux, Piercarlo Rossi, and Andrea Violato. 2016. The European Taxonomy Syllabus: A multi-lingual, multi-level ontology framework to untangle the web of European legal terminology. *Applied Ontology* 11, 4 (2016), 325–375. https://doi.org/10.3233/AO-170174

[2] VS Anoop, S Asharaf, and P Deepak. 2016. Unsupervised concept hierarchy learning: a topic modeling guided approach. *Procedia Computer Science* 89 (2016), 386–394.

[3] Korinna Bade and Andreas Nürnberger. 2014. Hierarchical constraints - Providing structural bias for hierarchical clustering. *Machine Learning* 94, 3 (2014), 371–399. https://doi.org/10.1007/s10994-013-5397-9

[4] Gioele Barabucci, Angelo Di Iorio, Francesco Poggi, and Fabio Vitali. 2013. Integration of Legal Datasets: From Meta-model to Implementation. In *Proceedings of International Conference on Information Integration and Web-based Applications &#38; Services (IIWAS '13)*. ACM, New York, NY, USA, Article 585, 10 pages. https://doi.org/10.1145/2539150.2539180

[5] Holger Bast, Georges Dupret, Debapriyo Majumdar, and Benjamin Piwowarski. 2006. Discovering a Term Taxonomy from Term Similarities Using Principal Component Analysis. In *Semantics, Web and Mining*, Markus Ackermann, Bettina Berendt, Marko Grobelnik, Andreas Hotho, Dunja Mladenič, Giovanni Semeraro, Myra Spiliopoulou, Gerd Stumme, Vojtěch Svátek, and Maarten van Someren (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 103–120.

[6] Mark Belford, Brian Mac Namee, and Derek Greene. 2018. Stability of topic modeling via matrix factorization. *Expert Systems with Applications* 91 (2018), 159–169.

[7] Guido Boella, Luigi Di Caro, Michele Graziadei, Loredana Cupi, Carlo Emilio Salaroglio, Llio Humphreys, Hristo Konstantinov, Kornel Marko, Livio Robaldo, Claudio Ruffini, Kiril Simov, Andrea Violato, and Veli Stroetmann. 2015. Linking Legal Open Data: Breaking the Accessibility and Language Barrier in European Legislation and Case Law. In *Proceedings of the 15th International Conference on Artificial Intelligence and Law (ICAIL '15)*. ACM, New York, NY, USA, 171–175.

[8] Paolo Bouquet, Fausto Giunchiglia, Frank van Harmelen, Luciano Serafini, and Heiner Stuckenschmidt. 2003. C-OWL: Contextualizing Ontologies. In *The Semantic Web - ISWC 2003*, Dieter Fensel, Katia Sycara, and John Mylopoulos (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 164–179.

[9] Mírian Bruckschen, Caio Northfleet, DM Silva, Paulo Bridi, Roger Granada, Renata Vieira, Prasad Rao, and Tomas Sander. 2010. Named entity recognition in the legal domain for ontology population. In *In: 3rd Workshop on Semantic Processing of Legal Texts (SPLeT 2010)*. 16.

[10] MarÃŋa G. Buey, Angel Luis Garrido, Carlos Bobed, and Sergio Ilarri. 2016. The AIS Project: Boosting Information Extraction from Legal Documents by using Ontologies. In *Proceedings of the 8th International Conference on Agents and Artificial Intelligence*. 438–445. https://doi.org/10.5220/0005757204380445 Exported from https://app.dimensions.ai on 2018/08/19.

[11] Philipp Cimiano, Andreas Hotho, and Steffen Staab. 2004. Clustering concept hierarchies from text. In *Proceedings of the Conference on Lexical Resources and Evaluation (LREC)*. 1721–1724.

[12] Peter Derleder, Kai-Oliver Knops, and Heinz Georg Bamberger. 2008. *Handbuch zum deutschen und europäischen Bankrecht*. Springer Science & Business Media.

[13] Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics, 1535–1545.

[14] Douglas H Fisher. 1987. Knowledge acquisition via incremental conceptual clustering. *Machine learning* 2, 2 (1987), 139–172.

[15] Enrico Francesconi, Simonetta Montemagni, Wim Peters, and Daniela Tiscornia. 2010. Integrating a bottom–up and top–down methodology for building semantic resources for the multilingual legal domain. In *Semantic Processing of Legal Texts*. Springer, 95–121.

[16] Marian George and Christian Floerkemeier. 2014. Recognizing products: A per-exemplar multi-label image classification approach. In *European Conference on Computer Vision*. Springer, 440–455.

[17] Korhan Günel and Rıfat Aşlıyan. 2010. Extracting learning concepts from educational texts in intelligent tutoring systems automatically. *Expert Systems with Applications: An International Journal* 37, 7 (2010), 5017–5022.

[18] Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*. Association for Computational Linguistics, 539–545.

[19] Rinke Hoekstra, Joost Breuker, Marcello Di Bello, Alexander Boer, et al. 2007. The LKIF Core Ontology of Basic Legal Concepts. *LOAIT* 321 (2007), 43–63.

[20] Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of classification* 2, 1 (1985), 193–218.

[21] Anette Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*. Association for Computational Linguistics, 216–223.

[22] Huang-Cheng Kuo, Tsung-Han Tsai, and Jen-Peng Huang. 2006. Building a Concept Hierarchy by Hierarchical Clustering with Join/Merge Decision. In *Proceedings of the 9th Joint Conference on Information Sciences, JCIS 2006*, Vol. 2006.

[23] C.J. MacLellan, E. Harpstead, V. Aleven, and K.R. Koedinger. 2016. TRESTLE: A Model of Concept Formation in Structured Domains. *Advances in Cognitive Systems* 4 (2016), 131–150.

[24] Karteeka Pavan, Allam Appa Rao, and A V Rao. 2011. An Automatic Clustering Technique for Optimal Clusters. abs/1109.1068 (09 2011), 133–144.

[25] Cécile Robin, James O'Neill, and Paul Buitelaar. 2017. Automatic Taxonomy Generation - A Use-Case in the Legal Domain. *CoRR* abs/1710.01823 (2017). arXiv:1710.01823

[26] Jorge M Santos and Mark Embrechts. 2009. On the use of the adjusted rand index as a metric for evaluating supervised classification. In *International Conference on Artificial Neural Networks*. Springer, 175–184.

[27] Anne Schiller, Simone Teufel, and Christine Thielen. 1995. *Guidelines für das Tagging deutscher Textcorpora mit STTS.* Technical Report. Universitäten Stuttgart und Tübingen.

[28] Erich Schweighofer, Anton Geist, et al. 2007. Legal Query Expansion using Ontologies and Relevance Feedback.. In *LOAIT*. 149–160.

[29] Vi sit Boonchom and Nuanwan Soonthornphisaj. 2012. ATOB algorithm: an automatic ontology construction for Thai legal sentences retrieval. *Journal of Information Science* 38, 1 (2012), 37–51. https://doi.org/10.1177/0165551511426249 arXiv:https://doi.org/10.1177/0165551511426249

[30] Sebastian Stober and Andreas Nürnberger. 2011. An experimental comparison of similarity adaptation approaches. In *International Workshop on Adaptive Multimedia Retrieval*. Springer, 96–113.

[31] Pepijn R.S. Visser and Zhan Cui. 1998. Heterogeneous Ontology Structures for Distributed Architectures.

[32] Fabio Vitali and Flavio Zeni. 2007. Towards a country-independent data format: the Akoma Ntoso experience. In *Proceedings of the V legislative XML workshop*. Florence, Italy: European Press Academic Publishing, 67–86.

[33] Bernhard Waltl, Georg Bonczek, and Florian Matthes. 2018. Rule-based Information Extraction - Advantages, Limitations, and Perspectives. *Jusletter IT* (02 2018).

[34] Bernhard Waltl, Jörg Landthaler, and Florian Matthes. 2016. Differentiation and Empirical Analysis of Reference Types in Legal Documents.. In *JURIX*. 211–214.

[35] Minmei Wang, Bo Zhao, and Yihua Huang. 2016. PTR: Phrase-Based Topical Ranking for Automatic Keyphrase Extraction in Scientific Publications. In *International Conference on Neural Information Processing*. Springer, 120–128.

[36] Shuting Wang, Chen Liang, Zhaohui Wu, Kyle Williams, Bart Pursel, Benjamin Brautigam, Sherwyn Saul, Hannah Williams, Kyle Bowen, and C Lee Giles. 2015. Concept hierarchy extraction from textbooks. In *Proceedings of the 2015 ACM Symposium on Document Engineering*. ACM, 147–156.

[37] Radboud Winkels, Alexander Boer, Bart Vredebregt, and Alexander van Someren. 2014. Towards a Legal Recommender System.. In *JURIX*, Vol. 271. 169–178.

[38] Paul Zhang and Lavanya Koppaka. 2007. Semantics-based legal citation network. In *Proceedings of the 11th international conference on Artificial intelligence and law*. ACM, 123–130.

ACKNOWLEDGMENTS

The authors would like to thank Andreas Nürnberger and the anonymous referees for their valuable comments. The work is supported by Legal Horizon AG, Grant No.: 1704/00082.