A Service Architecture for AI-based Legal Knowledge Extraction*

Valerio Bellandi[1], Silvana Castano[1], Stefano Montanelli[1] and Davide Riva[1]

[1]*Università degli Studi di Milano*
Department of Computer Science
Via Celoria, 18 - 20133 Milano, Italy

Abstract

The paper presents a reference service architecture for legal knowledge extraction based on a combination of Natural Language Processing and Machine Learning techniques/services. A case-study as well as experimental results are presented based on a pilot dataset of civil court decisions in the framework of the NGUPP project funded by the Italian Ministry of Justice.

Keywords

Legal Knowledge Extraction, Natural Language Processing, Legal Knowledge Graph, Digital Justice.

1. Introduction

Legal documents constantly produced by Parliaments, Courts, and other institutional bodies constitute a prominent source of information and knowledge not only for legal actors like judges or lawyers, but also for general subjects like citizens or private and public organizations. To improve both efficiency and effectiveness of courthouses and legal record offices and to foster digital justice, a significant effort is being devoted in almost all countries to digital transformation projects, by developing legal information systems and modular architectures providing a variety of services for acquisition, management, classification, exploration, and retrieval of legal documents. In this context, knowing how to navigate the complex structure and content of legal documents is an arduous task, and the availability of a legal knowledge extraction service is not only desirable but even mandatory, to capture and formalize the features and variety of legal terminology into representative concepts enabling the retrieval of pertinent and relevant chunks of information within large corpora of legal documents.

In this paper, we present a reference service architecture for legal knowledge extraction based on a combination of Natural Language Processing and Machine Learning techniques/services, with application and experimentation on a pilot dataset of civil court decisions in the framework

SEBD 2023: 31st Symposium on Advanced Database System, July 02–05, 2023, Galzignano Terme, Padua, Italy

[*] Discussion paper.

[*] Corresponding author.

[†] These authors contributed equally.

valerio.bellandi@unimi.it (V. Bellandi); silvana.castano@unimi.it (S. Castano); stefano.montanelli@unimi.it (S. Montanelli); davide.riva1@unimi.it (D. Riva)

0000-0003-4473-6258 (V. Bellandi); 0000-0002-3826-2407 (S. Castano); 0000-0002-6594-6644 (S. Montanelli); 0009-0003-9681-9423 (D. Riva)

of the NGUPP project funded by the Italian Ministry of Justice. We also discuss some preliminary evaluations of the proposed legal knowledge extraction service on the EurLex dataset.

Related Work. Legal knowledge extraction relates to the extraction of terms, rules, and concepts from legal documents [1]. Competitions in this research field helped to develop methods for these and other tasks. COLIEE has addressed legal information extraction and entailment on case law and statutes [2]. TREC Legal Track and AILA (Artificial Intelligence for Legal Assistance) track have focused mainly on legal document retrieval [3, 4].

Compared with document retrieval, knowledge extraction from legal documents has received lower interest so far. The work in this direction has mainly favored the conceptualization of domain ontology models [5, 6, 7], and the use of ontologies and thesauri to extract specific kind of knowledge (e.g., abstract terms [8], or legal rules [9]). In text mining, knowledge extraction has traditionally been performed by frequency-based [10] and rule-based approaches [11]. Since their advent, transformer-based language models have been widely adopted for such task. For instance, [12] improves Named Entity Recognition using pre-trained BERT model [13]; [14] relies on BERT to extract topics and their associated terminology; [15] uses BART generative model [16] to extract $(subject, predicate, object)$ triples. Since the legal domain suffers a lack of annotated data, pre-trained language models can be effectively used in the context of Zero-Shot Classification (ZSC), namely the task of classifying data instances with labels that were never observed in the data [17]. While transformer models fine-tuned on Italian legal language have been developed (e.g., LamBERTa [18] and Italian Legal BERT [19]), we adopt Sentence-BERT pre-trained models [20] due to a preference for consistent sentence semantic representation over token representation.

2. The proposed service architecture

The modules/components of the proposed service architecture are shown in Figure 1.

A *data ingestion* layer is defined to acquire the corpus of legal documents that needs to be managed (e.g., law, judgements, sentences). Ingestion is executed as a stream operation, meaning that the documents of the corpus are acquired when they become available and they can be progressively added without system downtime. A storage layer is defined to maintain i) the document database for the raw *ingested documents* and corresponding texts; ii) the document *annotations* as well as the *index* system for full text, metadata, and annotation search; iii) the graph database for the *Entity Registry* (ER) to store a unique entry for the entities extracted from documents; iv) the *system logs* and related data to monitor the overall system. The storage layer exposes the ER APIs to manage both the entity types (the ER metamodel) and the entity instances as described in [21]. Document texts and metadata are stored in an ElasticSearch instance, while annotations in a SQL database as discussed in our previous work [22]. As a graph database for ER, we employed Neo4j.

The layers for *back-end components* and *front-end components* constitute the backbone of the proposed architecture. In back-end components, we distinguish modules for *document manager*, *service catalogue*, and *NLP services*. About the *document manager* modules, with their APIs, they

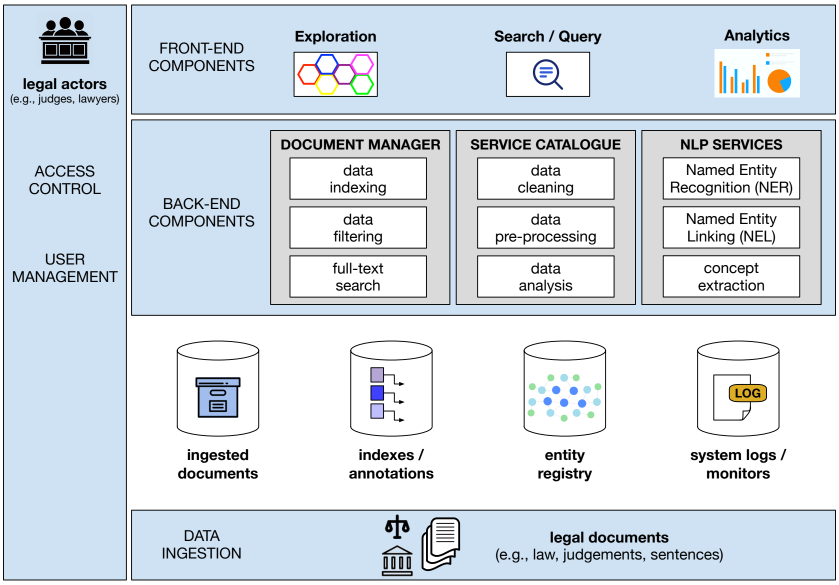

Figure 1: Service architecture for legal knowledge extraction.

can be considered proxies for client programs to index, filter, and fetch data from the storage. Configurable *service catalogue* modules are also defined for exploitation at ingestion time. The catalogue provides functionalities to process the incoming data and to create manipulated versions of the original documents through cleaning, pre-processing, summarization, and so on. The catalogue can also provide analysis functionalities over data, to be invoked for analytics purposes. Finally, the service catalogue can support orchestration functionalities to manage the workflow of articulated services. A set of *NLP services* is included in the back-end. They provide specific services according to the kind of mining operations that the system aims to support, like for example Named Entity Recognition (NER) and Linking (NEL), and concept extraction. A Kafka queue is created when a NLP service is invoked, with all the information needed to get the proper data. All the NLP services must expose standard APIs to be called by the system; they must read the Kafka queue and use the parameters found in it to obtain the input texts. At the end, they pass back their output to the document manager modules for storage in the annotation database and the entity registry.

In front-end components, we extend our previous work in [22] and we distinguish modules for *exploration*, *search/query*, and *analytics*. These modules expose APIs to enforce the interaction of users with the back-end components. Exploration allows to move from one document to another according to similarity-based criteria. The idea is to provide a service for browsing the corpus according to their common entities and/or concepts extracted by NLP services. Search/query allows to retrieve pertinent documents according to input entities/concept of interest provided by the final user. Analytics allows to examine the corpus through summary/statistical views built over data, such as for example the distribution of an entity or concept in the corpus, the shortest path (through documents) between given concepts or entities; the centrality of entities and concepts, and so on.

Finally, appropriate modules are included in the architecture to provide conventional *access control* and *user management* functionalities.

Some modules, like the ingestion and the document management, are executed in the back-end, in a transparent mode with respect to the user. A document becomes available for the front-end services when the ingestion stage is completed. Some other modules, like the NLP services and the front-end services, are *lively* invoked in response to a user request. In a typical scenario, the document manager modules are invoked to ingest the (set of) documents to import. Documents are stored *as is*, then cleaning and pre-processing are executed to extract and store cleaned copies upon which full text and metadata indexing are executed. NLP services are then invoked by front-end components to enforce the specific service functionality required by the final user. In this stage, the filtering module of the back-end can be invoked before the NLP service to define the subset of documents to use for satisfying the user request.

In the remaining of the paper, we present a knowledge extraction pipeline and an example of exploration service based on a case-study of Italian sentences.

3. The knowledge extraction service

The knowledge extraction service exploits the ingested documents to mine a set of featuring concepts that provide a topic-oriented description of their textual contents. The concepts extracted from the documents are organized in a graph, where a pair of similar concepts is linked by an edge. Each concept is also connected to the document portions from which the concept emerged, meaning that we can explore the pertinent document segments where a certain concept somehow occurs. Our solution exploits Natural Language Processing (NLP) techniques based on zero-shot learning and context-aware embedding models to enforce concept extraction. A detailed description of the proposed zero-shot learning approach to classification of legal documents is provided in [23]. In the following, we discuss how such an approach to knowledge extraction has been integrated as a pipeline in the infrastructure of Figure 1.

3.1. Data pre-processing

For knowledge extraction, the data pre-processing stage is based on a tokenization step, where the text of each ingested document d is split into a set of chunks. A *document chunk* k represents the text unit to consider for classification and it determines the granularity of the document that can be associated with a concept. We stress that the size of the document chunk should be large enough, so that the context can be captured, but not too much extended to avoid segments that are long to read and potentially noisy due to the presence of multiple concepts. In this paper, we choose to tokenize documents by defining a chunk for few sentence/phrase detected in a document, up to a maximum size of 512 words. This is particularly appropriate for legal actors (e.g., lawyers, practitioners) that are typically interested in retrieving precise document excerpts in which a given concept of interest appears and can be rapidly read/assimilated.

As a further pre-processing step, the terms appearing in document chunks are lemmatized and a vector-based representation of each document chunk is finally built. The use of embedding techniques to represent chunks allows to map the document contents on a semantic vector space where the similarity of two chunks can be measured by comparing the corresponding vector representations through a similarity metric (e.g., cosine similarity). For embedding construction, Sentence-BERT [20], a modification of the original BERT model based on siamese and triplets networks, is employed to derive a semantically meaningful embedding for a given sentence/phrase. As such, a document chunk is associated with a set of terms W_k therein contained. Any term is described as $w = (w_l, w_d, \bar{w})$, where w_l is the label of the term (i.e., the lemma), w_d is a description of the term meaning taken from a reference dictionary/vocabulary (e.g., WordNet), and $\bar{w}$ is the corresponding vector-based representation according to Sentence-BERT, respectively. A document chunk k has the form $k = (k_d, \bar{k})$, where k_d is the original textual content of the chunk and $\bar{k}$ is the corresponding vector-based representation calculated as the mean of term vectors $\bar{w}$ with $w \in W_k$. Embedding models have the capability to represent and compare the meaning of entire text blocks like document chunks. On such a target, context-aware embedding models fine-tuned on document similarity tasks, like Sentence-BERT, are appropriate. In the legal field, the phrase structure can be highly articulated, and some common terms can have a precise technical meaning when used in a court (e.g., citation, clemency, designation). Sentence-BERT can handle such a kind of situations, which may strongly deviate with respect to everyday conversations.

3.2. Concept extraction

The document chunks are exploited by zero-shot learning techniques to enforce a multi-label classification process with the aim at detecting a set of featuring concepts. Zero-shot learning is an unsupervised classification technique, characterized by the capability to enforce classification without requiring any pre-existing annotation of the considered documents.

Initially, a *seed knowledge* is defined as a set of textual descriptions, each one featuring a concept of interest, namely a *seed concept*, to consider for classification. Typically, for a seed concept, a basic, gross-grained description is provided as a short text (e.g., one or two phrases) or a list of keywords. As an example, for a seed concept about banking contract, a corresponding textual description used for embedding is bank deposit, safe deposit box, bank credit opening, bank advance, bank account, bank discount. Further concepts are derived from seed ones during the extraction process, and they usually provide a more fine-grained description of the concept instances occurring in the document chunks. A concept c, either seed or derived, is defined as a pair $c = (c_l, \bar{c})$, where c_l is a label featuring the meaning of the concept expressed in a synthetic and human-understandable way, and $\bar{c}$ is a vector-based concept representation. Each concept c is initially associated with the set of terms W_c extracted from the textual description of c. The vector concept $\bar{c}$ is built as the mean of the vectors of all the terms in W_c. Finally, the label c_l corresponds to the label w_l of the term $w \in W_c$, whose vector representation $\bar{w}$ is closest to the concept vector $\bar{c}$. Concept extraction is defined as a progressive, iterative process articulated in the following three steps:

Zero-shot classification. Given a set of concepts (i.e., the seed concepts at the beginning of the process), the document chunks are classified through zero-shot learning. A similarity measure σ, e.g. cosine similarity, is calculated over any pair of embeddings between chunks and concepts. A document chunk k is classified with the concept c when the similarity value satisfies $\sigma(\bar{k}, \bar{c}) \geq \alpha$, with α defined as a similarity threshold configured in the system. The value of α is empirically determined according to experimental results. In this paper, the value

$\alpha = 0.3$ is employed in the proposed case-studies and experiments.

Terminology enrichment. Given a document chunk k classified with the concept c, the terms in W_k are exploited for enriching the term set W_c. The idea is that the initial description of the concept c can become more detailed if we add terminology taken from chunks that are pertinent (i.e., classified) with c. This is done by summing, for each $w \in W_k$, similarities $\sigma(\overline{w}, \overline{c})$ and $\sigma(\overline{w}, \overline{K_c})$, where $\overline{K_c}$ denotes the average embedding of document chunks classified with c. Terms $w \in W_k$ satisfying a system-defined β similarity threshold are inserted in W_c.

Concept derivation. By enriching the term set W_c, it is possible that more fine-grained concepts emerge from c, and they can be generated as new concepts. The discovery of possible new concepts emerging from c is enforced by clustering the embedding vectors $\bar{w}$ of terms in W_c. The Affinity Propagation (AP) algorithm is adopted to this end, since it allows to detect the emergence of sub-groups of similar terms within W_c, without requiring to "a-priori define" the number of clusters to generate. A new concept c' is created for each cluster returned by AP on the terms W_c of a concept c. A link is defined between a concept c' and c to denote that c' is derived from c and they are somehow similar/related in content. The concept c is then updated since the terms in W_c can be changed due to enrichment. As a consequence, c_l and $\bar{c}$ are re-calculated.

The set of concepts obtained after derivation can trigger the execution of a new cycle based on the above three steps. New derived concepts can contribute to improve the classification of chunks with more fine-grained concepts. Further new concepts can be also discovered through a new execution of enrichment and derivation on the basis of a refined classification result. As such, concept extraction is characterized by a predefined endpoint condition based on a *termination threshold*. When the number of new concepts created in the derivation step is lower than the threshold, the concept extraction process is concluded. A final concept graph providing a topic-based description of the underlying document corpus is stored in the entity registry for subsequent exploitation by the front-end services. An example of concept graph extracted from a case-study of Italian legal documents will be discussed in Section 4.

4. Application to the Italian context and evaluation

In the following, we discuss some application examples and evaluation results by considering a corpus of Italian court decisions collected in the framework of the *Next Generation UPP* (*NGUPP*) project, funded by the Italian Ministry of Justice.

We consider a case-study about "unfair competition" as subject matter and we invoke our knowledge extraction service with the aim to explore the concepts extracted from the corpus on such a subject. The user can enforce a preliminary filtering step over the document metadata to select the set of court decisions to consider for concept exploration. The example is based on a dataset of 34 documents resulting from the following filtering operations: first level of judgment, judicial district in North-Western Italy, year of decision from 2008 onwards, subject matter corresponding to 172011 or 172012, that are subject codes related to unfair competition in the Italian law. In Figure 2, we show the concept graph returned by the knowledge extraction service for describing the filtered dataset on unfair competition. We note that most of the graph concepts pertain to the domain of trade justice (e.g., "consortium", "partnership", "transaction"), by also

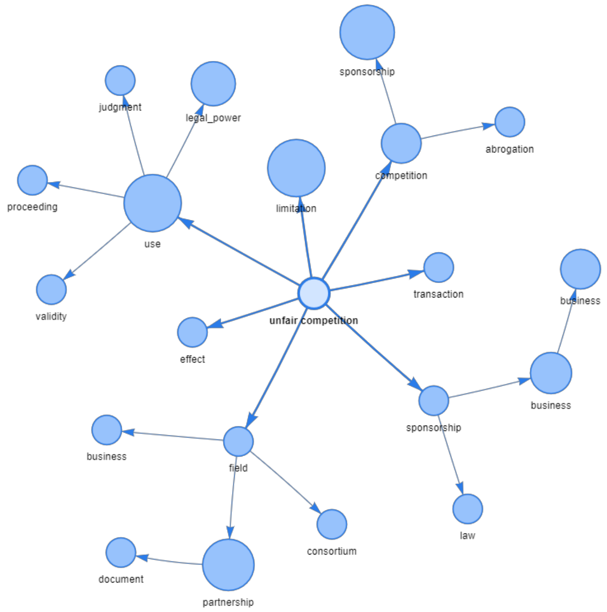

Figure 2: Example of concept graph returned by the knowledge extraction service for the case-study on "unfair competition". The size of a concept node is proportional to the number of document chunks classified with the concept. The original Italian labels of concepts have been translated into English for the sake of readability.

describing specific aspects concerned with unfair competition. Through links, it is possible to move from specific concepts (e.g., "sponsorship") to more general ones (e.g., "business"), and vice-versa. In the example, general concepts are usually associated with more chunks than specific concepts. We also note that some concept labels appear many times (e.g., "business", "sponsorship"), meaning that they refer to different senses of the concept label.

For evaluation of our concept extraction process, we consider EurLex57k, that is a dataset of 57,000 EU legislative documents annotated with labels representing entities, concepts, and topics from the EuroVoc thesaurus[1]. The goal of the evaluation is to assess whether our extracted concepts correspond with the labels of EuroVoc used for annotating the EurLex57k dataset. As a baseline, we consider BERTopic [14] since it is a topic modeling approach based on BERT and the mined topics can be straightforwardly compared to our extracted concepts. In Figure 3, we show the precision-recall curve obtained by our concept extraction service when various values of α and β thresholds are employed. We note that our solution outperforms the BERTopic baseline: despite a 0.05 decrease, precision remains higher than the baseline even when recall increases (i.e., when more concepts are extracted).

As a further experiment, we consider the results of the zero-shot classification and we evaluate the correspondence of our extracted concepts assigned to chunks w.r.t. the EuroVoc label assigned to documents. Results in terms of precision and recall are shown in Table 1

[1] https://eur-lex.europa.eu/browse/eurovoc.html?locale=en.

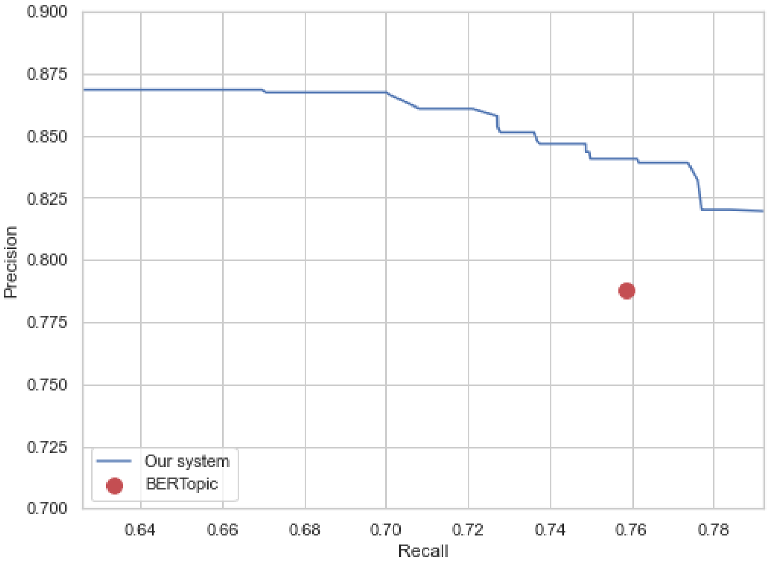

Figure 3: Precision-Recall curve for concept extraction on the EurLex57k dataset.

by providing mean and standard deviation at the document level. In the experiment, the following thresholds are set: $\alpha = \beta = 0.3$. We note that precision and recall of our concept extraction service are not only higher, but also significantly less variable than the ones obtained by BERTopic according to the standard deviation.

Model	Precision	Recall
Our system	**0.593 (0.061)**	**0.681 (0.078)**
BERTopic	0.455 (0.306)	0.422 (0.287)

Table 1
Mean (standard deviation) results for document classification.

5. Concluding remarks

In this paper, we presented a service architecture for legal knowledge extraction based on NLP services. A case-study has been presented by considering a dataset of Italian court decisions within the *NGUPP* project funded by the Italian Ministry of Justice. Preliminary results are promising. Ongoing activities as about the development of a Proof-of-Concept of the proposed architecture where a larger dataset of sentences will be considered from multiple legal-subject areas. The integration of more services is under development as well as the capability to orchestrate complex workflows where multiple services are involved. Moreover, future research work is about the comparison and possible extension of our NLP services through alternative mechanisms for document annotation (e.g., Semantic Role Labeling).

Acknowledgments

This work is partially supported by i) the Next Generation UPP project within the PON programme of the Italian Ministry of Justice, and ii) the project SERICS (PE00000014) under the MUR NRRP funded by the EU - NextGenerationEU.

References

[1] M.-F. Moens, C. Uyttendaele, J. Dumortier, Information extraction from legal texts: the potential of discourse analysis, International Journal of Human-Computer Studies 51 (1999) 1155–1171.

[2] J. Rabelo, M.-Y. Kim, R. Goebel, M. Yoshioka, Y. Kano, K. Satoh, Coliee 2020: methods for legal document retrieval and entailment, in: New Frontiers in Artificial Intelligence: JSAI-isAI 2020 Workshops, JURISIN, LENLS 2020 Workshops, Virtual Event, November 15–17, 2020, Revised Selected Papers 12, Springer, 2021, pp. 196–210.

[3] G. V. Cormack, M. R. Grossman, B. Hedin, D. W. Oard, Overview of the trec 2010 legal track., in: TREC, 2010.

[4] P. Bhattacharya, K. Ghosh, S. Ghosh, A. Pal, P. Mehta, A. Bhattacharya, P. Majumder, Fire 2019 aila track: Artificial intelligence for legal assistance, in: Proceedings of the 11th Annual Meeting of the Forum for Information Retrieval Evaluation, 2019, pp. 4–6.

[5] R. Hoekstra, J. Breuker, M. Di Bello, A. Boer, et al., The lkif core ontology of basic legal concepts., LOAIT 321 (2007) 43–63.

[6] G. Barabucci, A. Di Iorio, F. Poggi, F. Vitali, Integration of legal datasets: from meta-model to implementation, in: Proceedings of International Conference on Information Integration and Web-based Applications & Services, 2013, pp. 585–594.

[7] M. Palmirani, M. Martoni, A. Rossi, C. Bartolini, L. Robaldo, Pronto: Privacy ontology for legal reasoning, in: Electronic Government and the Information Systems Perspective: 7th International Conference, EGOVIS 2018, Regensburg, Germany, September 3–5, 2018, Proceedings 7, Springer, 2018, pp. 139–152.

[8] S. Castano, A. Ferrara, M. Falduti, S. Montanelli, Crime knowledge extraction: an ontology-driven approach for detecting abstract terms in case law decisions, in: Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law, 2019, pp. 179–183.

[9] M. Dragoni, S. Villata, W. Rizzi, G. Governatori, Combining nlp approaches for rule extraction from legal documents, in: 1st Workshop on MIning and REasoning with Legal texts (MIREL 2016), 2016.

[10] R. Mihalcea, P. Tarau, Textrank: Bringing order into text, in: Proceedings of the 2004 conference on empirical methods in natural language processing, 2004, pp. 404–411.

[11] G. Stanovsky, I. Dagan, Creating a large benchmark for open information extraction, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Austin, Texas, 2016, pp. 2300–2305. URL: https://aclanthology.org/D16-1252. doi:10.18653/v1/D16-1252.

[12] C. Liang, Y. Yu, H. Jiang, S. Er, R. Wang, T. Zhao, C. Zhang, Bond: Bert-assisted open-

domain named entity recognition with distant supervision, in: Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining, 2020, pp. 1054–1064.

[13] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[14] M. Grootendorst, Bertopic: Neural topic modeling with a class-based tf-idf procedure, arXiv preprint arXiv:2203.05794 (2022).

[15] G. Rossiello, F. Chowdhury, N. Mihindukulasooriya, O. Cornec, A. Gliozzo, Knowgl: Knowledge generation and linking from text, arXiv preprint arXiv:2210.13952 (2022).

[16] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, arXiv preprint arXiv:1910.13461 (2019).

[17] M.-W. Chang, L.-A. Ratinov, D. Roth, V. Srikumar, Importance of semantic representation: Dataless classification., in: Aaai, volume 2, 2008, pp. 830–835.

[18] A. Tagarelli, A. Simeri, Unsupervised law article mining based on deep pre-trained language representation models with application to the italian civil code, Artificial Intelligence and Law (2021) 1–57.

[19] D. Licari, G. Comandè, Italian-legal-bert: A pre-trained transformer language model for italian law, in: Companion Proceedings of the 23rd International Conference on Knowledge Engineering and Knowledge Management, Bozen-Bolzano (Italy), 2022.

[20] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, arXiv preprint arXiv:1908.10084 (2019).

[21] V. Bellandi, S. Siccardi, An Entity Registry: A Model for a Repository of Entities Found in a Document Set, Computer Science & Information Technology (CS & IT) 13 (2023).

[22] C. Batini, V. Bellandi, P. Ceravolo, F. Moiraghi, M. Palmonari, S. Siccardi, Semantic Data Integration for Investigations: Lessons Learned and Open Challenges, in: Proc. of the IEEE Int. Conference on Smart Data Services (SMDS), Chicago, IL, USA, 2021.

[23] V. Bellandi, S. Castano, P. Ceravolo, E. Damiani, A. Ferrara, S. Montanelli, S. Picascia, A. Polimeno, D. Riva, Knowledge-Based Legal Document Retrieval: A Case Study on Italian Civil Court Decisions, in: Proc. of the 1st Int. Knowledge Management for Law Workshop (KM4LAW), Bozen-Bolzano, Italy, 2022.