

An Indian Court Decision Annotated Corpus and Knowledge Graph

Pariskhit Kamat¹, Shubham Kalson¹, Suraj S¹, Pooja Harde^{1,*},
Nandana Mihindukulasooriya² and Sarika Jain^{1,*}

¹National Institute of Technology Kurukshetra, India

²IBM Research, Dublin, Ireland

Abstract

Document collection is increasing enormously in the legal domain, which requires automatic steps to analyze the data and curate the information from the same. Many challenges are being faced by the legal stakeholders to extract the information from the lengthy and unstructured court judgment documents relating to the main concepts, topics, and named entities in the documents. It has become an essential task in the current scenario to automate the information extraction process and store the documents in a properly structured format along with the different named legal entities for ease in the information extraction. In this paper, we introduce an annotated Indian Court Decision Document Corpus consisting of 10 coarse-grained classes and 30 fine-grained classes as a benchmark data set for constructing the knowledge graph. We also construct the Indian Court Case Documents' knowledge graph by utilizing a rule-based approach for Named Entity Recognition (NER) and Relation Extraction (RE). The results are evaluated against the proposed benchmark based on precision, recall, and F1 score and also qualitatively using SPARQL queries. The proposed approach gives a good F1 measure, though, further work is required to improve the recall.

Keywords

Entity Extraction, Relation Extraction, Knowledge Graph, Legal Domain.

1. Introduction

India's vast and complex legal system routinely creates and processes large volumes of legal documents. The limited knowledge of the general public in the field of law along with the complex language and legal terminologies makes it difficult for them to understand the ideas and information conveyed by legal documents. Even legal professionals who compile court judgment documents [1, 2, 3] find it cumbersome to go through long documents, understand and form opinions [4]. The existing portals retrieve the court decision documents either in PDF or unstructured format using basic keywords. To overcome the constraints of keyword-based search, semantic web and semantic search can be utilized. The semantic web[5] is a network in which the information is stored as knowledge graphs rather than a collection of documents linked using hyperlinks. Through semantic search, we will be able to focus on the intent and

Joint Proceedings of ISWC2022 Workshops: the International Workshop on Artificial Intelligence Technologies for Legal Documents (AI4LEGAL) and the International Workshop on Knowledge Graph Summarization (KGSUM) (2022)

*Corresponding author.

✉ pmharde29@gmail.com (P. Harde); nandana@ibm.com (N. Mihindukulasooriya); jasarika@nitkkr.ac.in (S. Jain)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

contextual meaning of the used keywords rather than completely relying on the keywords for information retrieval.

One of the major objectives of this work is to create an annotated data set for the Indian Court Decision Documents so that the machines can extract maximum information from the case documents and represent them in a uniform structured format with the help of Knowledge Graphs [6]. The work also explores the rule-based approach to extract and annotate the legal entities to construct a Knowledge Graph. The resultant Knowledge base will be a prodigious collection of interlinked case documents which can benefit the legal stakeholders. The primary step towards constructing the Knowledge Graph is the Information Extraction (IE) in which the various legal entities will be identified with the help of Named Entity Recognition (NER) and the relations among these entities will be extracted through Relation Extraction (RE).

The existing legal ontologies like JuDo [7], LKIF [8], LRI-Core [9] and so on are available for creating the legal domain knowledge bases and to serve the legal reasoning in the legal domain, these ontologies address legal entities and common sense entities at the very abstract level. Thus the already available ontologies do not fulfill the purpose of extracting the information from the court judgments like the Court, Jurisdiction under which the court can hear the case, type of evidence presented in the court hearing of the case, the origin of the case, case background, and so on. Other than ontologies there is a widely used XMLSchema defined as Akoma Ntoso¹ used for legal document structuring. The limitations with XML is they lack integration of heterogeneous data on the web and also does not provide the inference to the data which RDFS and OWL provides. Thus we move forward with the ontology which can provide the inference to the legal data for legal reasoning and semantics. NyOn[10] (Nyaya Ontology) is designed to primarily extract the relevant information from the court judgment documents and derive the relationship between this information making it available for different uses cases like legal reasoning, question-answering, legal analysis and so on. In paper [11], the JCO ontology for Indian Court Judgments is created but it lacks the reusability and publishing of the ontology. NyOn [10] Ontology has been chosen to be served as the metadata for Information Extraction. The second step is to store the extracted data in a graph database in form of triples to query the database for information retrieval.

The prominent contributions put forth by this paper are as follows:

- Creation of Indian Court Decision Documents Annotated Corpus guided by NyOn [10].
- Identify the legal entities and relations from the court decision documents using the Rule-Based Approach.
- Construction of Knowledge Graph from the extracted data.
- Evaluation based upon quantitative and qualitative approaches.

The remaining portion of the paper is compiled as follows. Section 2 examines the related works carried out in data set construction and the rule-based approach for information extraction. Section 3 focuses on Dataset Construction which includes the methodologies followed in creating and validating the data set. Section 4 discusses the construction of a Knowledge Graph using NER and RE through a rule-based approach. Section 5 sheds light on the evaluation results and Section 6 interprets the conclusions drawn from this work and provides insights on areas of improvement and ideas for future work.

¹<http://www.akomantoso.org/>

2. Related Work

Though the legal domain has been benefiting from the semantic web and ontology technologies in the past years, dedicated work on the Indian legal domain is yet to be developed. We have analyzed a few works done on ontology-based information retrieval and the same is discussed below.

Aboaoga et al. [12] and R.Alfred et al. [13] have proposed a Rule-Based approach for recognizing the named entity type (person names) for Arabic and Malay articles respectively.

Judith Jeyafreeda Andrew et al. [14] developed a system that helps journalists to recognize legal entities like names of people, organizations, roles, and functions. The author used 2 methods; first, Conditional Random Fields as a statistical method and another is the rule-based technique for generating language-specific regular expressions.

P. H. Luz de Araujo et al. [15] presents named entity recognition dataset for Brazilian legal documents. Along with the open domain tags such as persons, locations, time entities, and organizations, the dataset contains law and legal cases entities specific tags.

Based on the annotated corpus Prathamesh et al. [16] created baseline models for automatically predicting rhetorical functions in court documents. They also demonstrated the use of rhetorical roles to increase performance on summarization and legal judgement prediction tests.

Vladislav Korablinov and Pavel Braslavski [17] provided the first Russian knowledge base question answering (KBQA) dataset known as RuBQ. The high-quality dataset included 1,500 Russian questions of varied difficulty, their English machine translations, SPARQL queries to Wikidata, reference responses, and a Wikidata sample of triples comprising entities with Russian labels.

Elena Leitner et al. [18] developed data set for Named Entity Recognition in German legal documents. They manually annotated approx 67,000 sentences with 2 million tokens.

Riaz et al. [19] have discussed the differences between Hindi and Urdu NER and concluded that the NER computational models for Hindi cannot be applied to Urdu. They have also presented a NER algorithm rule-based Urdu that outperforms the models that use statistical learning.

Thomas et al. [20] have investigated natural language texts of domains lacking generic named entities labelled domain data sets. They created a hybrid NER system that combines rule-based deep learning with clustering-based techniques to enable the extraction of generic entities.

In a paper published by Crotti Junior et al. [21] the author discussed the difficulties they had and the progress they achieved when creating a knowledge graph-based search engine for Wolters Kluwer Germany's collection of German court case data.

Filtz et al. [22] highlights the data representation and search problems in the legal domain data. They suggested a method for representing Austrian legal information (legal standards and court rulings), and they demonstrated how to use such information to create a legal knowledge graph.

Breukers et al. [9] presented two legal core ontologies for law. The first was the outcome of Valente's Ph.D. thesis [23] called FOLaw. This core ontology, known as LRI-Core, is made up of five main sections (or "worlds"): roles, occurrences, physical, mental, and abstract classes.

Jain et al. [24] present the similar approach for extracting the NER and RE from the legal documents. For identifying the named entities, ontology presented in [10] is being used by

the author in the paper. Author used the JAPE rules for extracting the legal entities and legal documents are processed using the GATE tool which are later exported in the inline XML for RE.

A generic architecture for legal knowledge systems, as described by Hoekstra et al. in their publication [8], includes a legal core ontology (LKIF) that enables knowledge exchange between current legal knowledge systems. LKIF had two primary roles 1) translation of legal knowledge bases expressed in various representation formats and formalisms and 2) knowledge representation formalism that is a component of a wider architecture for creating legal knowledge systems.

Ceci et al. [7] introduces an OWL2 ontology library of legal knowledge that relies on the metadata contained in judicial documents known as JudO. The ontology addresses meaningful legal semantics at the same time retaining a strong connection to source documents (i.e. fragments of legal texts).

Thomas et al. [11] presents a legal case ontology named Judicial Case Ontology (JCO) that incorporates the concepts and relations existent in the legal domain cases including the related terms from a set of real-life judicial decisions. The ontology supports the extraction of taxonomic and non-taxonomic domain-specific relationships from e-judgments.

3. Dataset Construction

3.1. Dataset Description

For creating an indian legal corpus, the legal documents are collected from the 'Indian Kanoon'² website, an online search engine provided for Indian legal documents. The Python script used for scraping the dataset is given in the Github repository. For ease of processing, the collected PDF documents were converted to text format. The pre-processing such as sentence splitting, tokenization, and POS tags annotation using SPACY³ are performed on these text files data. To restrict the scope of the data we made use of the list of the competency questions such as:

1. List all the cases of month X.
2. List all the cases filed in the year X.
3. What are the total number of cases filed under case type 'criminal'?
4. List all the cases with X is a judge.
5. What is the count of cases with 'Appeal is accepted' as the judgment?
6. What is the date of judgement for the case X.
7. List all the cases filed under 'Appellant Jurisdiction'.
8. Petitioner Name with CASE NO.: X.
9. List all the cases involving X as one of the party.
10. Count of appeals 'rejected' by the judge X.

To address the scope of the data, the required legal terms were taken from NyOn [10] Ontology, a modular ontology to describe court judgments, and has been published adhering to the

²<https://indiankanoon.org/>

³<https://spacy.io/>

Semantic Web best practices and FAIR principles. Two semantic classes are defined to support domain-specific tags. That is, a coarse-grained class and a fine-grained class consisting of 10 and 30 attributes respectively. Coarse is a more general legal semantic class that includes Court, Party, CourtDecision, Document, Jurisdiction, Location, CaseType, Author, CourtOfficial, and DateOfJudgment classes.

The created dataset is the gold standard dataset with manually identified Named Legal Entities from the tokens and tagged with domain-specific tags using the CoNLL-2003 format. In CoNLL-2003 data files, there contains four columns separated by a single space. Each word of the sentence is added to a single line and each sentence is followed by an empty line. A word is an initial item on each line, followed by a part-of-speech (POS) tag, a syntactic chunk tag, and a named entity tag. The dataset is encoded in three different encodings in CoNLL-2003 format: BILOU ((B-Beginning, I-Internal, L-Last, O-outside, U-Unit), IOB (I-Inside, O-Outside, B-Begin) and IOBES (I-Inside, O-Outside, B-Begin, E-End, S-Single). It is to be noted that the syntactic tags are not considered for the preparation of the data set. While a named entity is a pronoun or noun, which usually refers to the name of a person, place, etc., legal entities are basically the legal terms from the legal documents that might be names of parties involved, document numbers, bench, the title of the legal document, etc. A total of four annotators have participated in the construction of the corpus. The manually developed dataset consists of a total of 50 legal documents with 80,733 rows of tokenized words and their corresponding annotated legal tags. Table 1 depict the count of the particular legal tag in the whole dataset (represented by #) for the corresponding coarse-grained and fine-grained classes.

Some example attributes of the coarse-grained class and fine-grained class are as follows:

Party The coarse-grained class *Party* PT contains fine-grained classes *Respondent* RES, *Appellant* APLT, *Plaintiff* PLNF, *Petitioner* PETR.

Ex. PETITIONER PT : B. SHANKRANAND PETR Vs.
RESPONDENT PT : COMMON CAUSE & ORS. RES DATE OF JUDGMENT:
11/03/1996 BENCH: RAMASWAMY, K. BENCH: RAMASWAMY, K. G.B. PAT-
TANAIAK (J).

CourtOfficial The coarse-grained class *CourtOfficial* CRTOF contains fine-grained classes *Investigator* INVG, *Solicitor* SOL and *Judge* JD.

Ex. DATE OF JUDGMENT: 29/04/1991 BENCH: KANIA, M. H. BENCH:
KANIA, M. H. JD VERMA, JAGDISH SARAN. JD (J) CRTOF
RAMASWAMI, V. JD (J) CRTOF

3.2. DataSet Validation and Publication

The constructed dataset is validated by an expert Mr. Vaibhav Vats, Advocate, Punjab and Haryana High Court, Chandigarh. The expert reported an annotation accuracy of 92%. It was observed that the tag SPECIAL LEAVE PETITION was wrongly annotated as PETITION. The

Table 1

Coarse-grained Classes (#: count of the tags in corpus)

SNo.	Coarse-grained	Fine-grained	#
1	Court CRT	Supreme Court SC , High Court HC , Metropolitan Court MTPC , District Court DC , Tribunal TRBL	155
2	Party PT	Respondent RES , Petitioner PETR , Appellant APLT , Plaintiff PLNF	100
3	CourtDecision CD	Judgment JDG , Order ORD	50
4	Document DOC	Petition PTN , Appeal APPL , CourtJudgement CRTJD , FIR FIR , Other OTR	51
5	Jurisdiction JURD	Original JUR-OGNL , Appellant JUR-APLT , Advisory JUR-ADVSY , Review JUR-REVV	17
6	Location LOC	Country CTY , State STE , District DST , Taluka TLKA , Place PLC	37
7	CaseType CTYP	Civil CIVL , Criminal CRNL	42
8	Author AUTH	Judge who delivers judgment	26
9	CourtOfficial CRTOF	Investigator INVG , Solicitor SOL , Judge JD	148
10	DateofJudgment DOJUD	Date of Judgment	50

expert noticed still more entities that should have been included and the list is not limited to: Party (Complainant, Defendant, Prosecution, and Accused for the criminal cases); Jurisdiction (Regional, Pecuniary, Writ, Special Leave Petition); Case Type (Matrimonial, Consumer, IPR); and more. The documents list can be very exhaustive if we want. The annotation of all these will be considered as future scope work.

The data set is published using FigShare⁴ with CC by 4.0 licence with the DOI:<https://doi.org/10.6084/m9.figshare.19719088.v4>

4. Knowledge Graph Construction

Knowledge graphs are network representations of real-world entities consisting of nodes, edges, and labels. Representing a copious collection of unstructured data using knowledge graphs will ease the process of abridging the facts and information from extensive documents. Though there are multiple approaches for constructing knowledge graphs from unstructured data, we have used the rule-based approach as it can closely simulate human intelligence and offers the flexibility to incorporate cognitive processes into machines. For extracting named entities and their relations, the rule-based approach uses regular expressions to identify various lexical patterns and trigger words.

4.1. Named Entity Recognition

The entity extraction process is carried out by referring to the NyOn [10] Ontology and a total of 10 named legal entities, namely *Party*, *Court*, *Date of Judgment*, *Court Official*, *Author*, *Location*, *Case Type*, *Court Decision*, *Jurisdiction* and *Documents* were identified as given in Table1.

⁴<https://figshare.com/>

Table 2

Sample Outputs from NER

NER SAMPLE OUTPUT		
KEWAL KRISHAN VS. STATE OF PUNJAB on 06/03/1962	-	CASE_NAME
KEWAL KRISHAN	-	PETITIONER
STATE OF PUNJAB	-	RESPONDENT
06/03/1962	-	DATE_OF_JUDGMENT

Table 3

Sample Outputs from RE

RE SAMPLE OUTPUT		
CASE	hasCaseId	196203KS1SC
196203KS1SC	hasCaseName	KEWAL KRISHAN VS. STATE OF PUNJAB on 06/03/1962
196203KS1SC	hasParty	PETITIONER
PETITIONER	hasName	KEWAL KRISHAN
196203KS1SC	hasParty	RESPONDENT
RESPONDENT	hasName	STATE OF PUNJAB
196203KS1SC	hasDate	06/03/1962

The scraped data from Indian Kanoon² is passed through Python rules which contain regular expressions that trigger target words. The major predicaments faced while coding the Python rules are the amorphous nature of the legal documents which made it difficult to code regular expressions that could fit the entire corpus. Despite the irregular structure and format, we were able to come up with reasonable rules that fit a decent cut of the corpus. Each case will be mapped to a central entity "CASE" in the knowledge graph. An entity "CASE_NAME" is formed with the help of three other identified entities, namely "Petitioner"/"Appellant", "Respondent" and "Date Of Judgment". And if none of the above three entities are identified, the "CASE_NAME" will be assigned with "CASE_NO" or "APPEAL_NO" respectively, subject to their identification in the document.

The output from the NER phase is stored in a single text file with the extracted token and its corresponding identified entity to pass to the Relation Extraction Phase for obtaining relations between the entities. The code and the output files are provided in the [Github repository](#).

4.2. Relation Extraction

The relation between the entities extracted in the NER phase are identified in this relation extraction phase using a small python script. The NyOn [10] is referred for identifying the various relationships between the extracted entities obtained in NER Phase. A total of 14 relations, namely *hasCaseName*, *hasParty*, *hasDate*, *hasYear*, *hasMonth*, *hasAppealNo*, *hasCase-Type*, *hasAuthor*, *hasCourtOfficial*, *hasJurisdiction*, *hasCourt*, *hasLocation* and *hasCourtDecision* are identified. For uniquely identifying each case, a new entity "CASE_ID" is generated by

Table 4
Named Entity Recognition Evaluation Metrics

S.No.	Type	Total Entities	Identified Entities	Correct Entities	Precision	Recall	F1
1	PARTY	100	100	100	1.00	1.00	1.00
2	COURT	155	43	42	0.98	0.27	0.42
3	DATE OF JUDGEMENT	50	50	50	1.00	1.00	1.00
4	COURT OFFICIALS	148	114	114	0.77	0.77	0.77
5	AUTHOR	26	25	25	0.96	0.96	0.96
6	LOCATION	37	28	26	0.92	0.70	0.79
7	CASE TYPE	42	42	39	0.92	0.92	0.92
8	COURT DECISION	50	49	49	0.98	0.98	0.98
9	JURISDICTION	17	15	15	0.88	0.88	0.88
10	DOCUMENTS	51	40	31	0.77	0.60	0.67

concatenating the year and month of judgment, an abbreviation of our system name(KS for Kanoon Sarathi) along with the serial number of the case in the current month, and the court abbreviation to which it belongs to(SC for Supreme Court, HC for High Court and DC for District Court). A new relation *hasCaseId* is also derived from the new entity "CASE_ID".

Since the output of the NER stage does not contain sentences, we use 'if' statements to annotate the relationships between the extracted entities. A sample output containing a few entities and relations from both the NER phase and RE phase corresponding to the case *KEWAL KRISHAN VS. STATE OF PUNJAB on 06/03/1962* is given in Table 3 The code used for relation extraction along with the output file containing identified relations(predicates) along with the corresponding entities(subject and object) is provided in the [Github repository](#).

5. Evaluation

5.1. Quantitative Evaluation

For quantitative evaluation of the rule-based Named Entity Recognition (NER) and Relation Extraction (RE) with respect to our data set, we use the metrics F1-Score, Recall(for measuring the reliability of the model in correctly identifying entity tags out of actually existing entity tags), and Precision (for measuring the reliability of the model in correctly identifying entity tags out of total identified entity tags). Table 4 shown below represents the Evaluation metrics of Named Entity Recognition and Table 5 depicts the Evaluation Metrics for Relation Extraction.

5.2. Qualitative Evaluation

For the Qualitative evaluation of the Knowledge Graph, 10 competency questions were formulated and the knowledge graph is queried using SPARQL to retrieve the relevant information. The sample queries based on competency questions are performed on the knowledge graph that are shown in figure 1. The list of the competency questions with the corresponding queries and their outputs can be found in the [Github repository](#).

Table 5
Relation Extraction Evaluation Metrics

S.No.	Type	Total Relations	Identified Relations	Correct Relations	Precision	Recall	F1
1	hasCaseName	50	50	50	1.00	1.00	1.00
2	hasParty	100	100	100	1.00	1.00	1.00
3	hasName	248	202	188	0.93	0.75	0.83
4	hasCaseNo	50	44	41	0.93	0.82	0.87
5	hasDate	50	50	50	1.00	1.00	1.00
6	hasYear	50	50	50	1.00	1.00	1.00
7	hasMonth	50	50	50	1.00	1.00	1.00
8	hasAuthor	26	25	24	0.96	0.92	0.93
9	hasJurisdiction	15	15	15	1.00	1.00	1.00
10	hasCourt	50	43	42	0.97	0.84	0.90
11	hasLocation	37	28	28	1.00	0.75	0.85
12	hasCaseType	42	42	42	1.00	1.00	1.00
13	hasCourtDecision	49	49	49	1.00	1.00	1.00
14	hasCourtOfficial	50	50	50	1.00	1.00	1.00

6. Conclusion and Future Scope

In this paper, we have presented a dataset for Knowledge Base construction in the Indian Legal domain. We have also discussed the modus operandi for constructing Knowledge Graph from the Indian Court Decision corpus through a rule-based approach. NyOn [10] Ontology was used as a reference for entity extraction and relation extraction with the help of which the triples were annotated. After triple generation, the RDF conversion process is followed using python script, and the same is stored in Apache Jena Fuseki.

The results derived were arguably good and comparatively better than the existing works using a rule-based approach, albeit we have identified numerous shortcomings which can be improved. In terms of future work, we plan to extend the dataset in two dimensions; one, add more documents to increase the size of the dataset which will provide a good sample for approaching the machine learning algorithms for extracting named entities, and second, to add more entities for annotating legal norms, solicitors, evidence and so on.

6.0.1. Acknowledgements

This work is supported by the IHUB-ANUBHUTI-IIITD FOUNDATION set up under the NM-ICPS scheme of the Department of Science and Technology, India. We thank Mr. Vaibhav Vats, Advocate, Punjab and Haryana High Court, Chandigarh for providing his valuable reviews of the data set.

Supplemental Material Availability: The code and the data set are available on the GitHub Repository with link: <https://github.com/semintelligence/KING>.

```
select ?CaseName where
{
  ?case nyon:hasYear "1996";
  nyon:hasCaseName ?CaseName.
}
```

(a) List all the cases from the year 1996.

```
select ?CaseName where
{
  {
    ?case nyon:hasParty party:UNION_OF_INDIA_OR_S;
    nyon:hasCaseName ?CaseName.
  }union
  {
    ?case nyon:hasParty party:Union_of_India_Ors;
    nyon:hasCaseName ?CaseName.
  }union
  {
    ?case nyon:hasParty party:UNION_OF_INDIA_AND_OR_S;
    nyon:hasCaseName ?CaseName.
  }
}
```

(c) List all the cases with Union of India as the party.

```
select (count(?CaseName) as ?TotalCases) where
{
  ?case nyon:hasCaseType nyon:Criminal;
  nyon:hasCaseName ?CaseName.
}
```

(b) Count of all the criminal cases.

```
select ?CaseName where
{
  {
    ?participants rdfs:label "VIVIAN BOSE"@en.
    ?case nyon:hasCourtOfficial ?participants;
    nyon:hasCourtDecision decision:rej;
    nyon:hasCaseName ?CaseName.
  }union
  {
    ?participants rdfs:label "V. BOSE"@en.
    ?case nyon:hasCourtOfficial ?participants;
    nyon:hasCourtDecision decision:rej;
    nyon:hasCaseName ?CaseName.
  }
}
```

(d) List all the appeals rejected by the judge V. BOSE

Figure 1: SPARQL Queries representing different competency questions

References

- [1] A. Elnaggar, C. Gebendorfer, I. Glaser, F. Matthes, Multi-task deep learning for legal document translation, summarization and multi-label classification, in: Proceedings of the 2018 Artificial Intelligence and Cloud Computing Conference, 2018, pp. 9–15.
- [2] O.-M. Sulea, M. Zampieri, S. Malmasi, M. Vela, L. P. Dinu, J. Van Genabith, Exploring the use of text classification in the legal domain, arXiv preprint arXiv:1710.09306 (2017).
- [3] N. Ramrakhiyani, S. Pawar, G. K. Palshikar, A system for classification of propositions of the indian supreme court judgements, in: Post-Proceedings of the 4th and 5th Workshops of the Forum for Information Retrieval Evaluation, 2013, pp. 1–4.
- [4] V. Malik, R. Sanjay, S. K. Nigam, K. Ghosh, S. Guha, A. Bhattacharya, A. Modi, Ildc for cjpe: Indian legal documents corpus for court judgment prediction and explanation, ????
- [5] S. Sharma, S. Jain, Comprehensive study of semantic annotation: Variant and praxis, Advances in Computational Intelligence, its Concepts Applications (ACI 2021) 2823 (2021) 102–116.
- [6] M. Dragoni, S. Villata, W. Rizzi, G. Governatori, Combining Natural Language Processing Approaches for Rule Extraction from Legal Documents: AICOL International Workshops 2015-2017: AICOL-VI@JURIX 2015, AICOL-VII@EKAW 2016, AICOL-VIII@JURIX 2016, AICOL-IX@ICAIL 2017, and AICOL-X@JURIX 2017, Revised Selected Papers, volume 10791, 2018, pp. 287–300. doi:10.1007/978-3-030-00178-0_19.
- [7] M. Ceci, A. Gangemi, An owl ontology library representing judicial interpretations,

Semantic Web 7 (2016) 229–253.

- [8] R. Hoekstra, J. Breuker, M. Di Bello, A. Boer, et al., The lkif core ontology of basic legal concepts., LOAIT 321 (2007) 43–63.
- [9] J. Breukers, R. Hoekstra, Epistemology and ontology in core ontologies: Folaw and lri-core, two, in: Proceedings of EKAW Workshop on Core ontologies [Internet]. Northamptonshire, UK: Sun SITE Central Europe, Citeseer, 2004.
- [10] S. Jain, P. Harde, N. Mihindukulasooriya, Nyon - a multilingual modular legal ontology for representing court judgments, in: International Semantic Intelligence Conference (ISIC 2022) held during May 17-19, 2022, Georgia Southern University (Armstrong Campus), Savannah, United States, 2022.
- [11] A. Thomas, S. S., A legal case ontology for extracting domain-specific entity-relationships from e-judgments, 2017.
- [12] M. Aboaga, Arabic person names recognition by using a rule based approach, Journal of Computer Science 9 (2013) 922–927. doi:10.3844/jcssp.2013.922.927.
- [13] R. Alfred, L. M. C. Leong, C. K. On, P. Anthony, Malay named entity recognition based on rule-based approach, International Journal of Machine Learning and Computing 4 (2014) 300–306.
- [14] J. J. Andrew, Automatic extraction of entities and relation from legal documents, in: Proceedings of the Seventh Named Entities Workshop, Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 1–8. URL: <https://aclanthology.org/W18-2401>. doi:10.18653/v1/W18-2401.
- [15] P. H. Luz de Araujo, T. de Campos, R. Oliveira, M. Stauffer, S. Couto, P. De Souza Bermejo, LeNER-Br: A Dataset for Named Entity Recognition in Brazilian Legal Text: 13th International Conference, PROPOR 2018, Canela, Brazil, September 24–26, 2018, Proceedings, 2018, pp. 313–323. doi:10.1007/978-3-319-99722-3_32.
- [16] P. Kalamkar, A. Tiwari, A. Agarwal, S. Karn, S. Gupta, V. Raghavan, A. Modi, Corpus for automatic structuring of legal documents, CoRR abs/2201.13125 (2022). URL: <https://arxiv.org/abs/2201.13125>. arXiv:2201.13125.
- [17] V. Korablinov, P. Braslavski, Rubq: A russian dataset for question answering over wikidata, CoRR abs/2005.10659 (2020). URL: <https://arxiv.org/abs/2005.10659>. arXiv:2005.10659.
- [18] E. Leitner, G. Rehm, J. M. Schneider, A dataset of german legal documents for named entity recognition, CoRR abs/2003.13016 (2020). URL: <https://arxiv.org/abs/2003.13016>. arXiv:2003.13016.
- [19] K. Riaz, Rule-based named entity recognition in urdu, in: Proceedings of the 2010 Named Entities Workshop, NEWS '10, Association for Computational Linguistics, USA, 2010, p. 126–135.
- [20] A. Thomas, S. Sangeetha, An innovative hybrid approach for extracting named entities from unstructured text data, Computational Intelligence 35 (2019) 799–826. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/coin.12214>. doi:<https://doi.org/10.1111/coin.12214>. arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1111/coin.12214>.
- [21] A. Crotti Junior, F. Orlandi, D. Graux, M. Hossari, D. O'Sullivan, C. Hartz, C. Dirschl, Knowledge graph-based legal search over german court cases, in: European Semantic Web Conference, Springer, 2020, pp. 293–297.

- [22] E. Filtz, Building and processing a knowledge-graph for legal data, 2017. doi:10.1007/978-3-319-58451-5_13.
- [23] A. Valente, Legal knowledge engineering: A modelling approach, volume 30, Penn State Press, 1995.
- [24] S. Jain, P. Harde, N. Mihindikulasooriya, S. Ghosh, A. Dubey, A. Bisht, Constructing a knowledge graph from indian legal domain corpus, in: International Workshop On Knowledge Graph Generation From Text (Text2kg) Co-located with the Extended Semantic Web Conference (ESWC 2022), CEUR Workshop Proceedings, volume 3184, 2022, pp. 80–93.