Knowledge Graph-based Thematic Similarity for Indian Legal Judgement Documents using Rhetorical Roles

Sheetal S, Veda N, Ramya Narasimha Prabhu, Pruthvi P and **Mamatha HR**
Department of Computer Science and Engineering, PES University, India
{sheetals410,vedan1408, ramrag0107, mail.pruthvip}@gmail.com
mamathahr@pes.edu

Abstract

Automation in the legal domain is promising to be vital to help solve the backlog that currently affects the Indian judiciary. For any system that is developed to aid such a task, it is imperative that it is informed by choices that legal professionals often take in the real world in order to achieve the same task while also ensuring that biases are eliminated. The task of legal case similarity is accomplished in this paper by extracting the thematic similarity of the documents based on their rhetorical roles. The similarity scores between the documents are calculated, keeping in mind the different amount of influence each of these rhetorical roles have in real life practices over determining the similarity between two documents. Knowledge graphs are used to capture this information in order to facilitate the use of this method for applications like information retrieval and recommendation systems.

1 Introduction

The Indian legal system is currently facing the problem of legal pendency owing to the large volume of cases that are filed each day. This is exacerbated by the lack of trained legal professionals and absence of good resources to aid legal experts. Thus, creating sound legal tech systems, especially ones aided with the advancements that have been seen by the field of artificial intelligence, is imperative for remedying the current situation.

Identifying similarity between documents isn't a new task, but identifying the similarity between legal text documents is challenging, understudied and quite essential. A mechanism that can identify the similarity of legal cases quite clearly would make a great backbone for a powerful information retrieval engine or a recommendation system that would greatly boost any legal expert's research and preparations.

Presently, popular legal information retrieval systems like `https://indiankanoon.org/` often use plain full-text similarity that doesn't make use of any kind of context, semantic or otherwise.

Through this work, the task of legal case similarity is explored using knowledge graphs. To accomplish this, the use of semantic segments present in case documents, identified using deep learning, and the jurisdiction of the case itself are implemented. This information is then used to calculate similarity scores for case documents. A knowledge graph is used to store the extracted information because of its ability to capture the relationships between the documents. The intent behind creating such a system is to provide a sturdy and reliable backbone to information retrieval engines that can be used by legal experts and laymen alike for research, preparation, study, etc.

The paper is structured as follows: The subsection 2 illustrates the related work of researchers, section 3 shows the framework. Section 4 explains the methodology, that includes, description of the dataset, data-preparation steps, similarity metrics and methods used and details of how the case documents are stored in a knowledge graph. Section 5 discusses the result and application of this work in the real world. Finally, section 6, concludes the paper with a brief view about future scope.

2 Related Work

2.1 Rhetorical Role Identification

The work in Bhattacharya et al. (2019) aims to use neural models or deep learning models for the "task of rhetorical role identification." There were some prior attempts made that relied on hand-crafted features which had a few disadvantages. It gave reliable results for a few domains only and it required legal knowledge which was expensive to get. Hence neural models were chosen as it does not rely on any hand-crafted features. The two neural models used for the task are the Hierarchical-BiLSTM model and Hierarchical-BiLSTM-CRF.

Proceedings of the 19th International Conference on Natural Language Processing (ICON), pages 154 - 160
December 15-18, 2022

The results showed that the Hierarchical-BiLSTM-CRF model performed a little better than the other. The performance improvement was not so significant because CRF was unable to learn the emission score and transition score well. This happened because the legal document consists of a large number of sentences and only a few of them were considered for training purposes.

The paper Kalamkar et al. (2022) offers a corpus of English-language court judgment papers that are divided into relevant and cohesive sections. Each of these elements is labeled with a selection from a list of rhetorical roles. Based on the annotated corpus, they create baseline models for automatically predicting rhetorical roles in legal documents. There are 26,304 sentences annotated with 12 different rhetorical roles in the produced corpus, which comprises 265 Indian legal texts annotated with rhetorical roles. A transformer-based baseline approach for automatically annotating legal texts with sentence-level RR is also described in the paper. Finally, the research demonstrates how rhetorical roles can be used to improve legal summarization.

Majumder and Das (2020) worked on using models like Random Forest, Universal Sentence Encoder, BERT, and ROBERTA for labelling the rhetorical roles (RR). Sixty legal case documents from the Supreme Court were considered for this task. Fifty case documents were used for training and 10 for testing. Among all the models tried, ROBERTA outperformed the others. The output of ROBERTA was sent to BiLSTM. Three different models based on ROBERTA were tried each with different epochs. The first model was trained for 13 epochs, the second for 15, and the third trained for 19 epochs. Among the three models, the one trained for 15 epochs outperformed with a better Macro F-Score. However, the model was unable to label some RRs accurately.

2.2 Knowledge Graphs

The data used in Dong et al. (2021) is created as a result of extraction from semi-structured web pages and the attributes and relationships from the text are extracted using a Bi-GRU model. The graph constructed is visualized for better understanding on the China judgment Online website. However, they do not experiment with case similarity itself as a feature.

Incorporating features relevant to the legal domain, this work Dhani et al. (2021) on case similarity and citation-linked prediction builds on the use of knowledge graphs in Natural Language Processing tasks. For unstructured text from a corpus of court cases, laws, and rulings in Indian courts, a legal knowledge graph is constructed to represent the entities of a document and the relationships between them. Latent Dirichlet Allocation (LDA) is applied to model the most relevant topics for the derived ontology. Graph neural network models are used to identify missing links in a case graph constructed, with citation and similarity as relations, that has keywords/phrases specific to legal practice as node attributes. The performance of the relational graph convolutional networks on both tasks is shown to be higher when trained on the feature set containing lawpoints.

The paper Zhao et al. (2022) on legal judgment prediction deals with determining the law article, charge, and term of penalty given a fact description using graph neural networks. Judicial document text from the CAIL2018 dataset represented as 6 kinds of graphs which include co-occurrence, pointwise mutual information, semantic (using cosine similarity) and distance-based (Euclidean, Manhattan, and Chebyshev) underwent information updation between the graphs using a graph convolutional network. The prediction model fed with a fusion of information from graph nodes and law articles surpasses the baseline models in accuracy, macro-precision, macro-recall, and macro-F1 scores. The proposed method has an edge owing to the legal text differentiation extractor based on graph attention networks.

Usage of knowledge graphs for legal tasks has been undertaken before in Cavar et al. (2018), Filtz (2017) and Mandal et al. (2017) but these approaches don't make use of semantic segments which provide essential context for identifying similarity. Law practitioners often use their knowledge of the semantic segments in legal cases implicitly to identify similar cases. This, therefore, provides a strong motivation for identifying rhetorical roles and using knowledge graphs to capture them.

Works like Dhani et al. (2021) also make use of metadata like judge, court, and date. Approaches such as these end up relying less on law and more on context provided by such metadata which introduces bias and makes for a technology that isn't developed with fairness and equity in mind.

Use of knowledge graphs as the preferred method of storing information is motivated by the fact that it opens up the avenue to develop faster information systems and recommendation systems using the same strategy as the basis.

2.3 Case Similarity

Mandal et al. (2017) aims to improvise the text-based methods that are used to compute the similarities between documents and tried topic modelling, word, and document embedding. Among the different approaches tried, embedding based methods outperform the others. These approaches were tried on different representations of the documents like whole document, paragraphs, summaries, and text around citations. With summaries topic modelling (LDA) performed the best and with the whole document as a method of representation, Doc2Vec outperformed the other methods considered. However, there were a few drawbacks with summaries as a method of representation used because the results depended on the quality of the summaries and not all summaries were of high quality.

In the paper Bithel and Malagi (2021), unsupervised algorithms are used to rank documents based on their similarity to the query and top x documents are termed relevant documents. Approaches used by them include TFIDF with cosine similarity, Word embeddings, best match 25, TFIDF and BM25, Rake + TF-IDF, and cosine similarity sentence BERT. However, they do not use any of the more complex deep learning-based approaches and the possibility of training a BERT model from scratch so that the semantic meaning of prior cases and queries can be used.

The work Ostendorff et al. (2021) on legal literature recommendation aids research for a particular case by retrieving other decisions covering the same topic or necessary background information. Divided into (1) text-based (baseline TF-IDF, word-vector-based and transformer-based), (2) citation-based and (3) hybrid, 27 methods are compared on the basis of document length, and citation count and recommendation coverage over two datasets containing 2964 US case law documents. The results point to fastTextLegal as the overall best performing method. It is observed that the performance of text-based approaches such as Paragraph Vectors and Longformers is adversely affected by increasing word count and that of citation-based methods such as DeepWalk and Poincaré by decreasing citation count. Hybrid methods offer broad coverage and overcome the limitations of single methods.

In this paper Bhattacharya et al. (2020), similarity computation methods for legal documents based on textual content as well as precedent citation networks are analyzed on a common dataset of 47 pairs of Indian Supreme Court case documents. It describes text-based similarity measures like Paragraph Links, FullText Similarity using Doc2vec and proposes a novel method that considers aggregated scores between thematic segments (facts, arguments, rulings, statutes, etc) and network-based metrics like Bibliographic Coupling, Co-citation, Dispersion and Node2Vec, a unique algorithm for graph embeddings. Compared on the basis of their Pearson correlation coefficient, Node2Vec, FullText, and Thematic Similarity show comparable results. It is noted that a higher score is obtained with a combination of the 2 methods.

Previous works present several methods and approaches for similarity between legal documents, predominantly using full-text or citation-based techniques. The objective of our proposed solution is to cater to a particular user, here, law practitioners, by modeling their requirements while retrieving similar cases. Similarity is calculated using TF-IDF with cosine similarity and represented using the weighted average score between various rhetorical roles by effectively capturing their relative relevance.

3 Proposed Methodology

The intent with this work was to create a framework that is robust, reliable and focused on equity without bias . Therefore, the proposed solution introduces a framework that operates on an input case judgment document and generates as output the most similar case documents. The database consists of documents from the ILDC (Indian Legal Document Corpus) dataset. Rhetorical roles are then identified from all documents, similarity scores computed between the segments and represented as a knowledge graph as depicted in Figure 1.

4 Implementation

4.1 Dataset Used

The ILDC dataset, introduced in Malik et al. (2021), is used for this work . The dataset was originally created for the purpose of legal judgment prediction and explanation and contains judicial summary documents for more than 35,000 Supreme Court cases. The dataset is annotated with legal expert

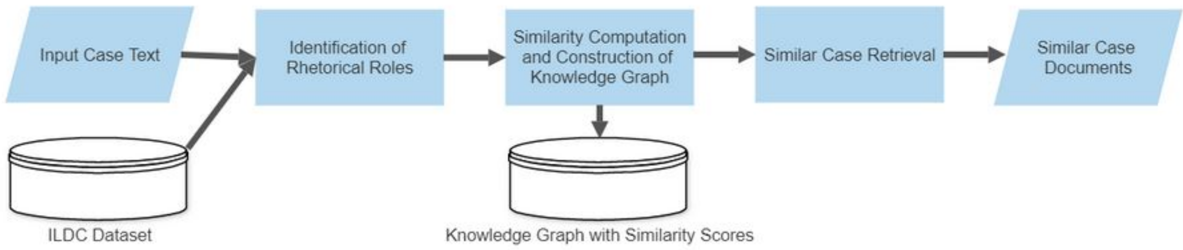

Figure 1: Diagrammatic Workflow of the Proposed Solution

guided explanations regarding the importance of different sentences in the judgment accompanied with the case. The motivation for using this dataset is the fact that it contains case documents without metadata like presiding judge and the place the case was filed which often introduce bias.

4.2 Identifying and labeling the relevant rhetorical roles

Rhetorical Role is also known as semantic segments. Each sentence in a legal case document can be assigned one of the predefined thirteen Rhetorical roles. Rhetorical role identification is a sentence classification task. The rhetorical role of a sentence does not only depend on the words in that sentence but also depends on the words from the sentence preceding and succeeding it. In other words, it depends on the context.

The impetus for identifying the rhetorical roles is based on the fact that the similarity between two cases is often ascertained based on the similarity between certain rhetorical roles.

To accomplish this task, the baseline method used by Kalamkar et al. (2022), which uses the SciBERT-HSLN architecture proposed by Brack et al. (2021) and is trained on the dataset contributed by the aforementioned work is adopted. Using this model, each line in the document is annotated with its corresponding rhetorical role as represented in Table 1.

4.3 Case Similarity

For a case document, once the rhetorical roles sentences are identified, all the sentences labeled with the same rhetorical role are grouped together. To determine how similar two case documents are, similarity scores between each of the thirteen rhetorical roles computed for both documents are being considered.

	Rhetorical Role	Label
1	Preamble	PREAMBLE
2	Facts	FAC
3	Ruling by Lower Court	RLC
4	Issues	ISSUE
5	Argument by Petitioner	ARG_PETITIONER
6	Analysis	ANALYSIS
7	Argument by Respondent	ARG_RESPONDENT
8	Statute	STA
9	Precedent Relied	PRE_RELIED
10	Precedent Not Relied	PRE_NOT_RELIED
11	Ratio of the decision	RATIO
12	Ruling by Present Court	RPC
13	None	NONE

Table 1: List of the Rhetorical Roles and their corresponding Labels

Similarity metrics like Jaccard Similarity, Euclidean Similarity, and Cosine Similarity were initially considered. To compute the similarity between documents, a few frequently used text embedding methods/techniques and deep learning-based textual similarity techniques were implemented.

4.3.1 Text Embedding Methods

TF-IDF: Term frequency is the normalized term count. Besides TF, another thing considered is how common a word is among all the documents and this is taken care of by the IDF. TF-IDF works by assigning more weightage to rare words and lesser weightage to commonly occurring words and, frequency of a word in a document relative to its frequency in the entire corpus can be found out.

Doc2Vec: Doc2vec generates numerical representations for sentences, paragraphs and documents. It represents the document into a vector of size 20 and hence there is no need to consider the average of word vectors to create document vectors.

Method	Precision	Recall	Micro F1
Jaccard	0.41	0.41	0.41
TF-IDF + Cosine	0.71	0.71	**0.71**
TF-IDF + Euclidean	0.43	0.43	0.43
Doc2Vec + Cosine	0.38	0.38	0.38
Doc2Vec + Euclidean	0.43	0.43	0.43
BERT + Cosine	0.47	0.47	0.47
BERT + Euclidean	0.43	0.43	0.43

Table 2: Scores for Similarity Computation Methods

4.3.2 Deep Learning-based Textual Similarity

BERT: BERT makes use of transformers, an "attention mechanism" to learn the contextual/semantic relations between words in the document. Since the transformer encoder reads the entire sequence at once (unlike directional models), it learns the context of the word based on the words towards its left and right. The BERT-base model fine-tuned for the NLI dataset was used to learn embeddings of the word in the document.

Legal experts were asked to rank a subset of documents from the ILDC dataset on the basis of their similarity to each other in order to understand how relevant the implemented systems are to the real-world requirement from the perspective of information retrieval systems of this kind. On comparing this with the implemented systems, TF-IDF, a text-based embedding method and cosine similarity as the similarity metric yielded the highest micro F1 score of 0.71, as shown in Table 2, and was chosen to compute the similarity.

4.4 Knowledge Graph Representation

The information regarding the case documents is stored in a knowledge graph as modelled in Figure 2. Knowledge graphs are abstract data structures that capture both the characteristics of entities and the relationships between them. They are formulated by: $G = (V, E)$

Legal documents are often related to each other thematically. Capturing this is likely to yield results that are better suited for real-world application. This intuition drives the use of knowledge graphs in this paper.

In the constructed knowledge graph, case documents are represented as nodes and each node has 13 attributes associated with them and they represent the 13 rhetorical roles that were identified and tagged in the documents. The value taken up by each attribute is the corresponding sentences in order to increase the robustness of the data captured.

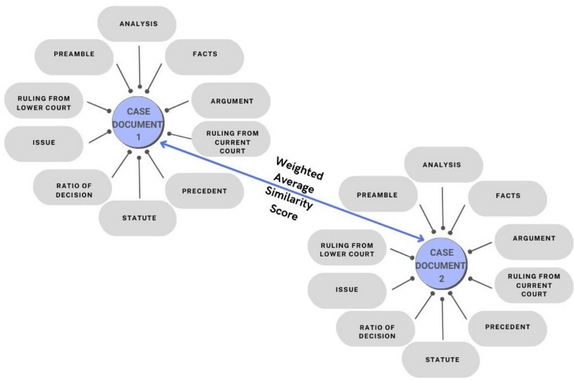

Figure 2: Knowledge Graph Representation

Rhetorical Role	Weight
ANALYSIS FAC	5
STA RPC RATIO ISSUE	4
RLC NONE	3
ARG_PETITIONER ARG_RESPONDENT PRE_RELIED	2
PREAMBLE	1

Table 3: Weights assigned to each Rhetorical Role

In practice, as pointed out by legal experts, different rhetorical roles have different amounts of relevance. To capture this as accurately as possible, the similarity scores of sentences in each individual rhetorical role in a case document are calculated using TF IDF. A weighted average of the same is then taken to obtain the corresponding edge weight.

The weights were decided by conferring with legal experts, as shown in Table 3. The importance of each rhetorical role while gauging the similarity between case descriptions as well as the accuracy of the baseline model used to identify them were considered in order to reduce the value of error.

The final weighted average similarity score is calculated as:

$$WeightedAverageSimilarityScore = \frac{\sum (Weights \times SimilarityScoreofRhetoricalRole)}{\sum Weights}$$

The weighted average similarity computation as per the priority order of the rhetorical roles resulted in a micro F1 score of 0.73, showing an improvement over the full-text methods.
The data structure containing the source and destination cases along with the similarity scores corresponding to the rhetorical roles thus obtained is stored both in a CSV and Neo4j, a graph database. The use of Neo4j as a database is dictated by its efficiency, ability to scale and ease of use. In order to ensure quick search and accurate retrieval, edges with low scores are dropped from the database. A snippet of nodes and their properties in the Neo4j database can be seen in Figure 3.

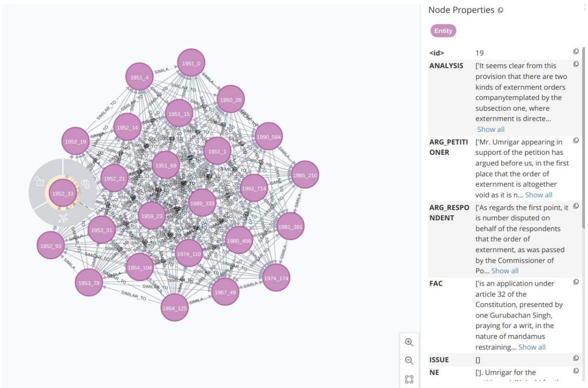

Figure 3: Representation of Knowledge Graph using Neo4j

5 Results

On comparing expert-anointed rankings for case documents based on their full-text similarity, TF-IDF with cosine similarity metric was observed to capture it most effectively with the highest micro F1 score of 0.71, amongst other methods such as BERT and Doc2Vec. Rhetorical roles were identified using the SciBERT-HSLN approach which achieved a micro f1 of 77.7 on hidden test data. Similarity scores were further enhanced to achieve a micro F1 score of 0.73, by integrating the aspect of thematic similarity using rhetorical roles. It was found to model expert-scores more accurately than the full-text methods alone. This information when represented in the form of a knowledge graph through the Neo4j database yielded an average query response time of 223 ms, showing an improvement nearly 10 times faster than retrieving the documents saved in a table. In a use case such as ours, it is of essence to note that the relationship between different documents is important. Only capturing course-grained details like document text and leaving similarity calculations to application logic leads to added computational costs incurred at every query, which can be avoided if this information is stored in a knowledge graph at the time that a new document is added to the database.

5.1 Applications in the Real World

5.1.1 Information Retrieval Systems

One of the most well-known applications of case similarity is information retrieval. Non-proprietary legal information retrieval systems like Indian Kanoon (https://indiankanoon.org/) have shortcomings which include the upper limit on the number of tokens that can be matched and the lack of robustness when the search is carried out.

An IR system backed by a database that stores documents like the one proposed in this paper can easily service multi-sentence queries by identifying the rhetorical roles present in the input and finding the corresponding weighted similarity scores.

This system can be further scaled over a large database with the help of clustering to reduce the number of similarity scores that need to be calculated.

5.1.2 Recommendation Systems

Recommendation systems can be used to suggest similar documents to law experts who are researching to make their preparation better. The usage of knowledge graphs makes it very easy to identify relevant similar documents faster.

The length of the feature vector for an item based collaborative filtering based recommender system with n users, m items, and c ratings is n; thus, the time complexity of the similarity computation is O (n). As a result, the overall temporal complexity is found to be $O(m^2n^2)$.

With the approach proposed by this work, the similarity computation has a time complexity of $O(|N|^2)$ (N being the number of nodes). The time complexity of the resulting recommendation system becomes $O(n * |N|^2)$.

6 Conclusion

This paper introduces a new method to compute and store similarity between judicial case documents. It identifies the rhetorical roles in case documents and leverages it to find the thematic similarity between the documents. The similarity scores thus obtained are stored in a knowledge graph along with the documents. In this representation the documents and their semantic segments are captured in the nodes and the similarity scores are represented as the edge weights. It is shown that this method can be used for building reliable information retrieval systems and recommendation systems.

For future work, using this data-structure for other legal tasks such as judgment prediction can be explored. It also remains to be determined if the task of explainability can be enhanced by the context provided by the identification of rhetorical roles. The weights in this work are determined using expert-rule methods. As an extension, different deep learning methods can be used to ascertain the optimal weights associated with the edges.

Acknowledgements

The authors thank Deeksha Kankane, a law student from Narsee Monjee Institute of Management Studies, Navi Mumbai who assisted this research by providing insights pertaining to the legal domain. The valuable technical and legal guidance provided by the professionals part of the team at Agami, India is gratefully acknowledged.

References

Paheli Bhattacharya, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2020. Methods for computing legal document similarity: A comparative study. *arXiv preprint arXiv:2004.12307*.

Paheli Bhattacharya, Shounak Paul, Kripabandhu Ghosh, Saptarshi Ghosh, and Adam Wyner. 2019. Identification of Rhetorical Roles of Sentences in Indian Legal Judgments.

Shivangi Bithel and Sumitra S. Malagi. 2021. Unsupervised Identification of Relevant Prior Cases. *CoRR*, abs/2107.08973.

Arthur Brack, Anett Hoppe, Pascal Buschermöhle, and Ralph Ewerth. 2021. Sequential Sentence Classification in Research Papers using Cross-domain Multitask Learning. *CoRR*, abs/2102.06008.

Damir Cavar, Jon Herring, and Alison Meyer. 2018. Law Analysis using Deep NLP and Knowledge Graphs. In *Proceedings of the LREC 2018 "Workshop on Language Resources and Technologies for the Legal Knowledge Graph"*.

Jaspreet Singh Dhani, Ruchika Bhatt, Balaji Ganesan, Parikshet Sirohi, and Vasudha Bhatnagar. 2021. Similar Cases Recommendation using Legal Knowledge Graphs. *CoRR*, abs/2107.04771.

Biao Dong, Haoze Yu, and Haisheng Li. 2021. A Knowledge Graph Construction Approach for Legal Domain.

Erwin Filtz. 2017. Building and Processing a Knowledge-graph for Legal Data. In *The Semantic Web*, pages 184–194, Cham. Springer International Publishing.

Prathamesh Kalamkar, Aman Tiwari, Astha Agarwal, Saurabh Karn, Smita Gupta, Vivek Raghavan, and Ashutosh Modi. 2022. Corpus for Automatic Structuring of Legal Documents.

Soumayan Bandhu Majumder and Dipankar Das. 2020. Rhetorical Role Labelling for Legal Judgements Using ROBERTA. In *FIRE*.

Vijit Malik, Rishabh Sanjay, Shubham Kumar Nigam, Kripabandhu Ghosh, Shouvik Kumar Guha, Arnab Bhattacharya, and Ashutosh Modi. 2021. ILDC for CJPE: Indian legal documents corpus for court judgment prediction and explanation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4046–4062, Online. Association for Computational Linguistics.

Arpan Mandal, Raktim Chaki, Sarbajit Saha, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2017. Measuring Similarity among Legal Court Case Documents. In *Proceedings of the 10th Annual ACM India Compute Conference*, Compute '17, page 1–9, New York, NY, USA. Association for Computing Machinery.

Malte Ostendorff, Elliott Ash, Terry Ruas, Bela Gipp, Julian Moreno-Schneider, and Georg Rehm. 2021. Evaluating document representations for content-based legal literature recommendations. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, pages 109–118.

Heng-Ru Zhang, Fan Min, Zhi-Heng Zhang, and Song Wang. 2019. Efficient collaborative filtering recommendations with multi-channel feature vectors. *International Journal of Machine Learning and Cybernetics*, 10.

Qihui Zhao, Tianhan Gao, Song Zhou, Dapeng Li, and Yingyou Wen. 2022. Legal Judgment Prediction via Heterogeneous Graphs and Knowledge of Law Articles. *Applied Sciences*, 12:2531.