CSE 6242: Analyzing U.S. Case Law with NLP and Data Visualization - Team 21

Paul Hutchison, Emre Duman, Bipin Koirala, Hung-Yu Shih, Mahdi Ghanei

1 INTRODUCTION

Legal research plays a crucial role in the practice of law, enabling legal professionals to interpret statutes, understand judicial precedents, and construct well-informed arguments. Thorough legal research ensures accuracy in legal decision-making, helps predict case outcomes, and enhances access to relevant legal information. As the volume of legal documents continues to grow, traditional research methods become increasingly time-consuming and inefficient. The need for more advanced, technology-driven approaches to legal research has never been greater, as legal professionals seek ways to navigate complex legal texts with greater speed and precision.

This project aims to develop an interactive data analytics and visualization framework for U.S. case law using the Caselaw Access Project dataset. By integrating text mining, natural language processing, and network analysis, the project seeks to extract legal trends, key arguments, and citation patterns to streamline legal research. Unlike traditional keyword-based legal research methods, our approach combines semantic analysis with interactive tools such as graphs and timelines to reveal deeper insights into judicial decision-making.

2 PROBLEM DEFINITION

Legal research is traditionally conducted manually by reading case law documents, searching legal databases, and relying on legal experts to interpret and summarize court decisions. Some legal analytics tools exist, but they often lack interactive visualization, focus primarily on keyword search rather than deep semantic understanding. The main limitation is the lack of scalable and automated tools that integrate NLP and data analytics to analyze legal texts beyond simple keyword searches.

Our key innovations include:

- Using LLM for text analysis for case categorization and complexity estimation
- Using similarity embedding to create a network graph for visualizing cases along with KNN search
- Visualizing different case stats using choropleth map
- Predict winning party using LLM model

The methods that we employed are focused around achieving these key innovations as optimally as possible. We will briefly explore the high level of what we are trying to achieve. For the case categorization, we are exploring the use of different LLMs to classify cases and estimate case complexity. We are currently using Google LLMs to achieve this. The similarity search task is done using familiar sentence embeddings that are then used primarily by the visualization scripts to create complex network interactions. The winning party predictions will also play a role here as we will be able to use this for filtering the data. We will be using a Legal-BERT model to perform this task. Finally we will create a choropleth map using data directly in manner that allows the user to explore different jurisdictions quickly.

3 LITERATURE SURVEY

This literature review aims to explore prior research on text mining in legal analytics, case law visualization, and predictive modeling of judicial decisions. [2] explored current trends in utilizing machine learning, deep learning and natural language processing techniques with the aim of predicting judicial decisions. We believe it provides a starting point to address questions such as which NLP and ML techniques are most used to predict judicial decision and what quality measures are used to validate decision prediction. Implementing some of these advanced models may necessitate access to high-quality annotated datasets and significant compute power. Our project could mitigate this by starting with simpler models or focusing on specific legal domains to reduce data complexity.

Studies like [13] employ network science methods to analyze citation patterns among legal cases, revealing underlying structures and the predictability of judicial decisions. While this study provides a robust network analysis, it may not fully integrate textual content analysis of case law. Our project could improve upon this by combining network metrics with natural language processing to capture both structural and semantic dimensions of legal documents.

[10] provides a framework to work with text data and ways to visualize them. We believe this paper serves as a direct reference for our project. While this paper specifically studies geological hazard documents, the techniques to visualize text data remain the same and carries over to different domains.

Building on previous studies, our project seeks to advance the integration of natural language processing (NLP), machine learning, and legal data analytics to enhance legal research. [3] demonstrate the effectiveness of NLP models in predicting judicial decisions, analyzing how textual features of case law contribute to predictive accuracy. While their approach focuses on the European Court of Human Rights, our project will apply similar methods to U.S. case law, leveraging the Caselaw Access Project dataset ([3]). [16] further support this direction by employing NLP and machine learning techniques to predict the outcomes of appeal decisions in Germany's tax law system, showcasing the potential for automated legal decision analysis ([16]). Additionally, [4] introduce a knowledge-driven legal service platform that integrates NLP and machine learning for legal document analytics, streamlining the organization, retrieval, and classification of legal texts ([4]). By leveraging these insights, our project aims to develop an interactive data analytics and visualization framework that enhances legal research by combining predictive modeling, text mining, and structured legal document management.

[18] introduces a topological learning approach that combines text analysis with graph-based representations of legal cases to predict judgments from the European Court of Human Rights, outperforming traditional NLP models by leveraging case interconnections.

[7] proposes a framework for leveraging ML and text mining to classify homicide-related court documents from the Caselaw Access Project, introducing a crime dictionary and a two-phase ML model to enhance police education and crime scene analysis.

[14] tackles the challenge of identifying semantic relationships in legal citation graphs by applying ML to annotate links between statutory provisions and precedents, enhancing the understanding of legal networks.

[5] introduce a graph-based framework for measuring legal document similarity. They constructed a network that incorporates not only case citations but also the hierarchy and citations of statutes. The similarity score is then calculated based on the structural similarity of the two documents within the network.

[11][12] explore various unsupervised methods for legal document similarity. They experimented with combinations between eight document representation techniques (whole document, summary, catchphrase and others) and seven vector representations (TF-IDF, word2vec, doc2vec, BERT and others). Their results show that using the whole document with doc2vec yields the best performance.

[1] researches and compares deep learning methods with classical ML approaches for predicting judicial decisions in cases. The main result of the paper found that CNN+BiLSTM approaches for prediction outperformed classical ML approaches by a significant margin in all metrics. This difference is even large when applying feature selection. This indicates that neural network approaches are well suited for this task.

[15] explores the use of large language models (LLMs) for processes such as precedent case prediction and case outcome prediction. They found that their models outperformed more generally trained LLMs and the LawLLM had relatively high prediction accuracy. This indicates that pre trained LLMs are a useful tool for the task at hand when attempting to predict precedent case precedent also.

[8] investigates uses and issues with visualization of legal data specifically through the lens of global economic law. The conclusions are that visualizations make the law more accessible and understandable to layman and professional people alike. The authors note that one of the biggest issues in the space is monopolization of data, but CAP resolves this issue for this application.

4 PROPOSED METHOD

4.1 Dataset

The dataset we are using is a subset of the original US case law dataset dataset that consists of 6.7 million cases. We are primarily focusing on the cases related to Georgia state for text analysis via LLMs. The dataset using can be found in Hugging Face: https://huggingface.co/datasets/free-law/Caselaw_Access_Project

4.2 Graph Visualization

In order to create a network of related legal cases we created a network graph visualization based off engineered subsets of the data. This engineering of the data is done primarily through the similarity search function to create links between the cases, then the winning party prediction computations and case type classification to create filters for the data.

The network visualization is intuitive to a user once that are viewing it. They are able to select a "seminal" case from some initially shown cases and then a graph of its k-nearest neighbors are shown using the similarity search functionality. A user is then able to filter this information intuitively using a dropdown menu, and the nodes are appropriately colored to indicate different classifications depending on what the filter system in use is. Further there is an ability to click into a case to get more information about the data such as the ID, name, or the opinion. This set of features allows for a user to sufficiently navigate the data without getting overwhelmed by the different things that can be done in the visualization, ensuring that he data is the foremost focus.

We will go on in later sections to explain the data and algorithmic innovations in greater depth. This visualization in iteself is innovative relative to the results found in the literature review as many of the existing tools kept the data analysis in purely numerical results without an associated visualization. Figure 1 demonstrates our graph visualization after our analysis.

4.3 Choropleth & KDE Visualization

To visualize the distribution of legal cases across U.S. states, we first extracted relevant information from the Caselaw Access Project dataset hosted on HuggingFace. We removed irrelevant columns and generated a filtered dataset containing only the jurisdiction, year, and number of cases. Due to memory constraints, it was not feasible to load the entire dataset into memory at once; hence, we processed the data sequentially and stored the filtered results incrementally.

During this process, we identified a data gap: the state of Alabama was missing from the HuggingFace version of the dataset. To address this, we accessed the Case.law website directly and retrieved the missing cases by volume, subsequently appending them to our existing filtered dataset.

Once we had complete data for all states, we generated a U.S. choropleth map to visualize the number of cases by state for a selected year (see Figure 2).

To provide a smoothed visualization of case count trends over time, we implemented a Kernel Density Estimation (KDE) plot based on the processed `.csv` data. The KDE was computed using the *Epanechnikov* kernel, defined as:

$$K(u) = \begin{cases} \frac{3}{4}(1-u^2) & \text{if } |u| \le 1 \\ 0 & \text{otherwise} \end{cases}$$

Given the observed case years $\{x_1, x_2, \ldots, x_n\}$ weighted by case counts, the density estimate at a point x is:

$$\hat{f}_h(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

where h denotes the bandwidth. In our visualization, the bandwidth was fixed at $h = 10$. Users can select a state from a dropdown menu, and the corresponding density curve for that state's case counts over time is rendered directly (see Figure 3).

4.4 Similarity Search

Instead of relying on traditional keyword-based methods for similarity search, we decided to use a vector-based approach using sentence embeddings from neural network models. We believe that contextualized vector embeddings can produce more accurate and relevant search results.

For generating sentence embeddings, we use two different neural network models, legal-BERT and BGE.

Legal-BERT [18] is a domain-specific BERT model pre-trained on legal corpora. It is tailored to understand legal terminology and context, making it more effective for tasks involving legal texts compared to general-purpose language models. For this task, we used the legal-bert-base-uncased model.

BGE (BAAI General Embeddings) [17] is a family of BERT-like models developed by the Beijing Academy of AI and Hugging Face, designed to create general text embeddings for downstream tasks. For this task, we used the bge-base-en-v1.5 model.

Since legal case texts often contain thousands of tokens, which is too long and computationally expensive for the models, we simplify the process by using only

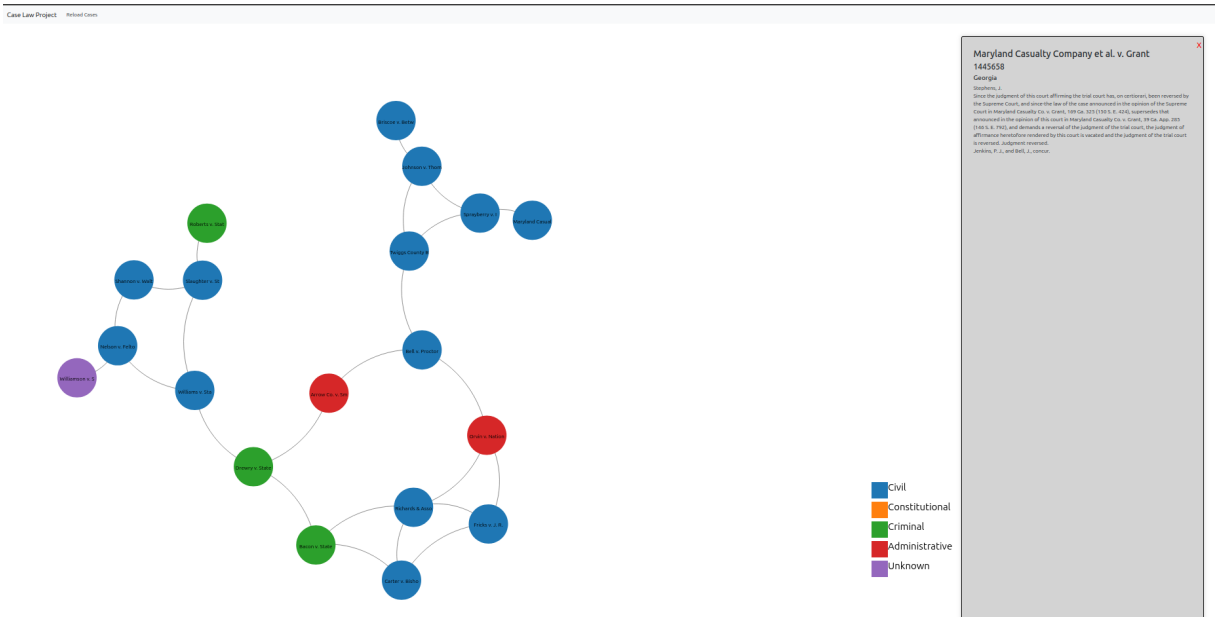

Figure 1: Network graph visualization

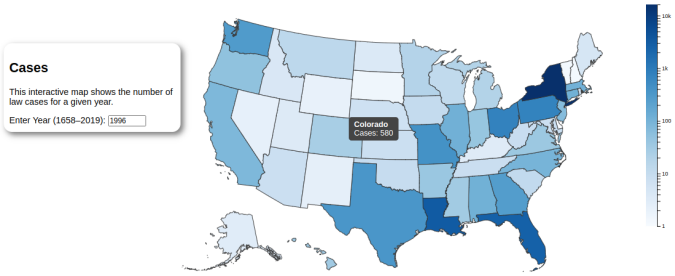

Figure 2: U.S. Choropleth map showing case counts by state for a selected year.

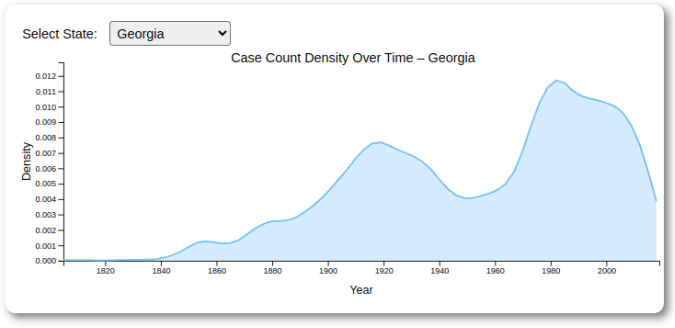

Figure 3: Kernel Density Estimate of Case Counts Over Time for a Selected U.S. State

the first 512 tokens as input. This portion usually captures the main idea, making it suitable for generating embeddings for our search database. In addition, to evaluate the performance of the search results, we also implemented a few baselines using Bag-of-Words, TF-IDF and Doc2Vec.

For the search itself, we use the K-nearest-neighbor (KNN) algorithm to find the K most similar cases based on Euclidean Distance. To speed up the process, we integrate FAISS (Facebook AI Similarity Search)[6], which supports efficient similarity searches on large datasets.

The search process works as follows: the user inputs a query, which is then processed through the BGE or Legal-BERT model to obtain its embedding. The KNN algorithm then retrieves the top K most similar search results in the datasets.

4.5 Case Categorization and Complexity

For each case, the record contains the several pages of opinions on the case and its details discussed at the time. Using a Large Language Model (LLM), we attempt to categorize the case into one of the four types of case law, namely, criminal, civil, bankruptcy, and appeal. Furthermore, we use LLMs to estimate the complexity of the particular case based on the numerical scale (with 1 indicating simple and 10 indicating most complex) using the presented opinions document.

This enables to do further analysis and correlate different aspects of the cases with each other.

We are currently using LLMs provided by Google to accomplish this.

4.6 Winning Party Prediction

We initially retrieved case records from the Hugging Face Caselaw Access Project dataset. A total of 889 cases from Alaska were used for this task. We extracted the CONCLUSION section from each case's text using a regular expression and captured additional context by extracting the next four sentences after the FACTS header. Rows lacking a CONCLUSION section were subsequently removed from the dataset. The extracted conclusion and context were then concatenated into a unified input for labeling the outcome using our pretrained Legal-BERT[18] model, which predicted the outcome as either "appellant" or "appellee." This preprocessing approach was adopted from [9]. Following this initial labeling, we removed both the conclusion and FACTS-related sections from the original text, producing a refined version of the case texts. We then used this modified text to train a GRU-based document encoder with a classifier, leveraging DistilBERT-based sentence embeddings to represent case documents. To mitigate class imbalance, oversampling was performed using a WeightedRandomSampler during training on the full dataset. Finally, we experimented with a Transformer Encoder-based model using the same DistilBERT sentence embeddings to further explore alternative architectures for legal outcome prediction. Evaluation F1 score and accuracy for both models were computed.

Model	Sentence Embedding	Accuracy	Macro F1 Score
GRU – Bidirectional	DistilBERT	81%	47%
Transformer Encoder	DistilBERT	74%	51%

Table 1: Performance metrics for winning party prediction using different architectures with DistilBERT sentence embeddings.

5 EVALUATION

5.1 Similarity Search

To evaluate the performance of our similarity search method, we constructed several baseline methods to assess how well our current search approach performs.

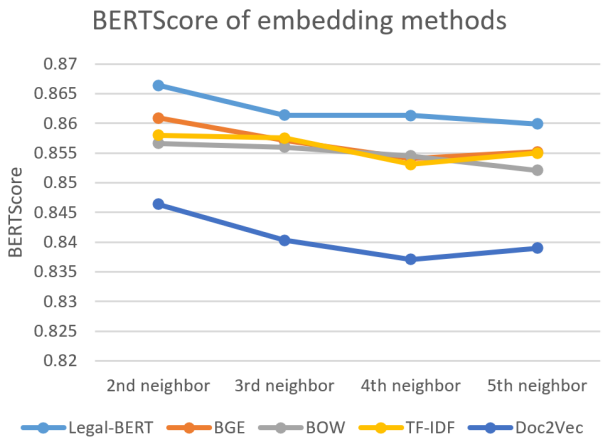

Figure 4: BERTScore of different embedding methods

These baselines use simpler embeddings, including Bag-of-Words, TF-IDF, and Doc2Vec. The performance of each method was calculated using the average BERTScore of the K search results retrieved by the same query set. This comprehensive comparison allows us to better understand the strengths and limitations of our current system in relation to these diverse baselines. The results are shown in Table 2 and Fig. 4.

In the experiment, the query set consisted of the first 50 cases in the Georgia state caselaw data, and K was set to 5. The Nth neighbor in the table represents the average BERTScore of the Nth closest result to the query in terms of Euclidean distance, retrieved by the KNN algorithm. Since the closest result is always the query itself, we excluded the first neighbor in the figure for a clearer comparison.

As we can see in the results, embeddings generated by two large transformer models, Legal-BERT and BGE, surpassed all baselines and achieved the best performance overall. We believe this is because the models acquired strong semantic understanding during the pretraining process, which makes them well-suited for this task. Since Legal-BERT has been fine-tuned on legal documents, it performs better than the BGE model, which is trained for general text embeddings. Among the baselines, BOW and TF-IDF achieved similar performance, but Doc2Vec performed significantly worse. We believe this is because caselaw texts are too long, making it difficult for the Doc2Vec algorithm to learn a good representation.

Method	1st neighbor	2nd neighbor	3rd neighbor	4th neighbor	5th neighbor	Avg
Legal-BERT	1	0.8664	0.8614	0.8613	0.8599	0.8898
BGE	1	0.8609	0.8571	0.8541	0.8552	0.88546
BOW	1	0.8566	0.856	0.8545	0.8521	0.88384
TF-IDF	1	0.858	0.8575	0.8531	0.855	0.88472
Doc2Vec	0.9934	0.8464	0.8403	0.8371	0.839	0.87124

Table 2: BERTScore with top 5 neighbors and average score for each embedding method.

5.2 Case Categorization and Complexity

While it is impossible to exactly evaluate the performance of our case categorization and complexity analysis since there are no labels in the dataset, we attempt to evaluate the performance of our prediction via using a smart thinking LLM on a subset of the dataset to provide us with more accurate predictions. Note that it is difficult to quantitatively evaluate this particular sub-task due to the subjectivity present in the opinions document for each case. We used the Gemini-2.0-Falsh-Lite API to extract the case category (Civil, Criminal, Administrative, and Constitutional), subcategories (theft, family law, etc.), as well as the numerical complexity metric for each case. We had 40 cases batched together for each API call to facilitate the analysis. We found that roughly 58% of the cases were civil, 33% were criminal and the rest were administrative or constitutional.

5.3 Winning Party Prediction

The performance metrics for winning party predictions are shown in Table 1. The GRU–Bidirectional model achieved an accuracy of 81% with a macro F1 score of 47%, indicating strong performance on the dominant appellee class but insufficient sensitivity to the minority appellant class. In contrast, the Transformer Encoder model achieved a slightly lower accuracy of 74% while obtaining a higher macro F1 score of 51%, suggesting that its self-attention mechanism better balances performance across classes. This result indicates a trade-off between overall accuracy and balanced performance, as measured by the macro F1 score. The relatively low macro F1 scores for both architectures highlight the challenge of identifying the minority class accurately. Overall, these findings underscore the necessity to further address class imbalance and refine model architectures for improved legal outcome prediction.

6 CONCLUSIONS AND DISCUSSION

In this project, we developed a data analytics and visualization framework combining natural language processing, machine learning, and interactive visualizations to enhance U.S. case law analysis. Leveraging Legal-BERT and BGE embeddings, our similarity search outperformed traditional methods like TF-IDF and Doc2Vec, demonstrating the importance of domain-specific models. The intuitive network visualization effectively enabled interactive exploration of legal case relationships, highlighting semantic connections among cases. Additionally, our choropleth and KDE visualizations offered clear geographical and temporal insights into case distributions, adding crucial context to traditional analysis methods.

Predictive modeling tasks, including winning party prediction and case categorization, showed promising accuracy but highlighted challenges due to class imbalance and subjective case complexity. The GRU-Bidirectional and Transformer Encoder models performed well overall, yet achieving balanced sensitivity across classes remains an important area for improvement.

Future directions include refining NLP models through further fine-tuning, addressing class imbalance with advanced sampling methods, and expanding the framework to cover broader jurisdictions. Ultimately, this integrated approach holds significant promise for streamlining legal research, making complex judicial data more accessible and interpretable for professionals and the public alike.

7 CONTRIBUTIONS

All members contributed equally. See Gantt Chart attached separately for planned actions.
Link to the Gantt Chart

REFERENCES

[1] Shakeel Ahmad, Muhammad Zubair Asghar, Fahad Mazaed Alotaibi, and Yasser D. Al-Otaibi. 2022. A hybrid CNN+BILSTM deep learning-based DSS for efficient prediction of judicial case decisions. *Expert Systems with Applications* 209 (2022), 118318. https://doi.org/10.1016/j.eswa.2022.118318

[2] Olga Alejandra Alcántara Francia, Miguel Nunez-del Prado, and Hugo Alatrista-Salas. 2022. Survey of text mining techniques applied to judicial decisions prediction. *Applied Sciences* 12, 20 (2022), 10200.

[3] Nikolaos Aletras, Dimitrios Tsarapatsanis, Daniel Preoţiuc-Pietro, and Vasileios Lampos. 2016. Predicting judicial decisions of the European Court of Human Rights: A natural language processing perspective. *PeerJ computer science* 2 (2016), e93.

[4] Valerio Bellandi, Silvana Castano, Stefano Montanelli, and Stefano Siccardi. 2025. Streamlining Legal Document Management: A Knowledge-Driven Service Platform. *SN Computer Science* 6, 2 (2025), 1–17.

[5] Paheli Bhattacharya, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2020. Hier-SPCNet: A Legal Statute Hierarchy-based Heterogeneous Network for Computing Legal Case Document Similarity. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 1657–1660. https://doi.org/10.1145/3397271.3401191

[6] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The Faiss library. (2024). arXiv:2401.08281 [cs.LG]

[7] Rsha Mirza Alaa Bafail Somayah Albaradei Ezdihar Bifari, Arwa Basbrain and Wadee Alhalabi. 2024. Text mining and machine learning for crime classification: using unstructured narrative court documents in police academic. *Cogent Engineering* 11, 1 (2024), 2359850. https://doi.org/10.1080/23311916.2024.2359850

[8] Manxia Huang. 2022. The Application of Data Visualization in Economic Law Under the Background of Big Data. In *Frontier Computing*, Jason C. Hung, Neil Y. Yen, and Jia-Wei Chang (Eds.). Springer Nature Singapore, Singapore, 450–457.

[9] Sahan Jayasinghe, Lakith Rambukkanage, Ashan Silva, Nisansa de Silva, and Amal Shehan Perera. 2022. Legal Case Winning Party Prediction With Domain Specific Auxiliary Models. In *Proceedings of the 34th Conference on Computational Linguistics and Speech Processing (ROCLING 2022)*, Yung-Chun Chang and Yi-Chin Huang (Eds.). The Association for Computational Linguistics and Chinese Language Processing (ACLCLP), Taipei, Taiwan, 205–213. https://aclanthology.org/2022.rocling-1.26/

[10] Ying Ma, Zhong Xie, Gang Li, Kai Ma, Zhen Huang, Qinjun Qiu, and Hui Liu. 2022. Text visualization for geological hazard documents via text mining and natural language processing. *Earth Science Informatics* (2022), 1–16.

[11] 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* (Bhopal, India) *(Compute '17)*. Association for Computing Machinery, New York, NY, USA, 1–9. https://doi.org/10.1145/3140107.3140119

[12] Arpan Mandal, Kripabandhu Ghosh, Saptarshi Ghosh, and Sekhar Mandal. 2021. Unsupervised approaches for measuring textual similarity between legal court case reports. *Artif. Intell. Law* 29, 3 (Sept. 2021), 417–451. https://doi.org/10.1007/s10506-020-09280-2

[13] Enys Mones, Piotr Sapieżyński, Simon Thordal, Henrik Palmer Olsen, and Sune Lehmann. 2021. Emergence of network effects and predictability in the judicial system. *Scientific reports* 11, 1 (2021), 2740.

[14] Ali Reza Sadeghian, Laksshman Sundaram, Daisy Zhe Wang, William F. Hamilton, Karl Branting, and Craig Pfeifer. 2018. Automatic semantic edge labeling over legal citation graphs. *Artificial Intelligence and Law* 26 (2018), 127 – 144. https://api.semanticscholar.org/CorpusID:3614985

[15] Dong Shu, Haoran Zhao, Xukun Liu, David Demeter, Mengnan Du, and Yongfeng Zhang. 2024. LawLLM: Law Large Language Model for the US Legal System. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24)*. ACM, 4882–4889. https://doi.org/10.1145/3627673.3680020

[16] Bernhard Waltl, Georg Bonczek, Elena Scepankova, Jörg Landthaler, and Florian Matthes. 2017. Predicting the outcome of appeal decisions in germany's tax law. In *Electronic Participation: 9th IFIP WG 8.5 International Conference, ePart 2017, St. Petersburg, Russia, September 4-7, 2017, Proceedings 9*. Springer, 89–99.

[17] Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-Pack: Packaged Resources To Advance General Chinese Embedding. arXiv:2309.07597 [cs.CL]

[18] Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. 2018. Legal Judgment Prediction via Topological Learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (Eds.). Association for Computational Linguistics, Brussels, Belgium, 3540–3549. https://doi.org/10.18653/v1/D18-1390