NITS_Legal at SemEval-2023 Task 6: Rhetorical Roles Prediction of Indian Legal Documents via Sentence Sequence Labeling Approach

Deepali Jain, Malaya Dutta Borah and **Anupam Biswas**
Department of Computer Science and Engineering
National Institute of Technology Silchar, India
jaindeepali010@gmail.com,{malayaduttaborah,anupam}@cse.nits.ac.in

Abstract

Legal documents are notorious for their complexity and domain-specific language, making them challenging for legal practitioners as well as non-experts to comprehend. To address this issue, the LegalEval 2023 track proposed several shared tasks, including the task of Rhetorical Roles Prediction (Task A). We participated as NITS_Legal team in Task A and conducted exploratory experiments to improve our understanding of the task. Our results suggest that sequence context is crucial in performing rhetorical roles prediction. Given the lengthy nature of legal documents, we propose a BiLSTM-based sentence sequence labeling approach that uses a local context-incorporated dataset created from the original dataset. To better represent the sentences during training, we extract legal domain-specific sentence embeddings from a Legal BERT model. Our experimental findings emphasize the importance of considering local context instead of treating each sentence independently to achieve better performance in this task. Our approach has the potential to improve the accessibility and usability of legal documents.

1 Introduction

Legal case documents are typically quite lengthy, often spanning many pages, which can make it time-consuming for legal practitioners and academics to read them in their entirety. In many cases, these professionals may only need to access specific portions of a document, such as the facts of the case or the arguments put forward by the parties involved (Jain et al., 2021d). However, legal case documents are often unstructured and lack clear section headings, unlike research papers or books. This can make it difficult for readers to quickly and efficiently locate the information they need.

The lack of structure in legal case documents can be particularly problematic for legal practitioners who are trying to build a strong legal argument. Without the ability to easily navigate and access relevant information, lawyers may struggle to build a coherent case that is based on sound legal principles and precedents.

To address this challenge, researchers and practitioners are exploring a range of approaches to structuring legal case documents and making them more accessible to readers. This includes the development of new tools[1] and technologies that can automatically extract key information from legal documents, such as the parties involved, the legal issues at stake, and the arguments put forward by each side (Farzindar and Lapalme, 2004; Polsley et al., 2016). By leveraging such tools, legal practitioners and academics can more quickly and efficiently access the information they need to build strong legal arguments and advance the field of law.

Rhetorical role labeling of sentences is a technique that can help legal practitioners quickly comprehend the structure and specific components of a legal case document (Teufel and Moens, 2002). This method involves identifying the semantic function associated with each sentence in the document. Formally, rhetorical role labeling refers to the process of classifying each sentence in a legal document based on its role in the overall document (Saravanan et al., 2008). By understanding the specific function of each sentence, legal practitioners can more easily identify the relevant portions of the document and extract the information they need to build a strong case. Such kind of upstream tasks are also helpful for performing downstream tasks such as summarization (Bhattacharya et al., 2019b). This can save valuable time and improve the efficiency of legal research and analysis.

To facilitate research in this area, the organizers of LegalEval 2023 have proposed a task of Rhetorical Labeling. The dataset provided for this task includes 247 training document-summary pairs, 30 development document-summary pairs, and 50 doc-

[1] https://tax-graph.273ventures.com/

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 751–757
July 13-14, 2023 ©2023 Association for Computational Linguistics

uments for testing. The sentences present in these documents are classified into 13 different rhetorical role classes: Preamble (PREAMBLE), Facts (FAC), Ruling By Lower Court (RLC), Issues (ISSUE) Argument by Petitioner (ARG PETITIONER), Argument by Respondent (ARG RESPONDENT), Analysis (ANALYSIS), Statute (STA), Precedent Relied (PRE RELIED), Precedent Not Relied (PRE NOT RELIED), Ratio of the decision (Ratio), Ruling By Present Court (RPC), None (NON). For more details about the task, please refer to the overview paper (Modi et al., 2023).

In this work, a detailed experimental study is conducted to solve the problem of rhetorical role labeling, by considering both sentence level as well as sentence sequence level classification approaches. Moreover, the utilization of legal domain-specific sentence embeddings is also considered in this work so that better model training is possible. From the experimental study it has been identified that domain-specific embeddings along with local context of sentence sequences are important for achieving improved performances in this task. Such an approach can significantly improve the comprehension of lengthy legal documents for legal practitioners as well as other readers. Our code is publicly available[2].

The rest of the paper is organized as follows: Section 2 presents the related works for the rhetorical roles prediction task. Section 3 describes our method used for performing rhetorical roles prediction. Section 4 presents the experimental results along with a detailed discussion. Finally, Section 5 concludes the paper with future research directions.

2 Related Works

Legal documents play a critical role in our society, as they provide the foundation for laws and regulations that govern our behavior. However, these documents can be difficult to read and understand, even for legal experts. In the recent years, there has been growing interest in the legal specific tasks such as Rhetorical labeling (Bhattacharya et al., 2019b; Malik et al., 2021a), legal document summarization (Jain et al., 2020, 2021c,b, 2022, 2023a,b), court judgment prediction (Chalkidis et al., 2020; Malik et al., 2021b; Niklaus et al., 2022) and so on. There have been several comparative analysis works performed using the legal documents to understand the behavior of these lengthy documents (Bhattacharya et al., 2019a; Jain et al., 2021a; Satwick Gupta et al., 2022).

Due to the progress of deep learning techniques, researchers have started employing these methods to analyze Indian court judgments for rhetorical labeling tasks as well. Farzindar and Lapalme (Farzindar and Lapalme, 2004) as well as Hachey and Grover (Hachey and Grover, 2006) have used the concept of rhetorical roles to generate summaries of legal texts. This approach involves identifying the various roles played by different segments of the text. By understanding these roles, the researchers were able to create condensed summaries that captured the main points of the text while maintaining its overall structure and coherence (Saravanan et al., 2008). Recently, Bhattacharya et al. (Bhattacharya et al., 2019b) have proposed BilSTM-CRF model to automatically assign rhetorical roles to the sentences of an Indian Case Judgement document. In another work, Malik et al. (2021a) have constructed a corpus of rhetorical roles (RR) and annotated it with 13 different detailed roles. Furthermore, they have proposed a multitask learning pipeline for identifying rhetorical roles. In another work done by Kalamkar et al. (2022), authors have created a larger rhetorical role dataset as compared to the dataset created by Malik et al. (2021a). The authors created a baseline system using the SciBERT-HSLN architecture (Brack et al., 2021).

In this work also, we utilize Indian Legal BERT (Paul et al., 2022) to create the embeddings of sentences in a sequence preserving the local context, followed by feeding them to a BiLSTM-based model for identifying the rhetorical roles.

3 System Description

The task of rhetorical role labeling can be modeled in several different ways. Keeping in mind the specific characteristics of legal documents, we set out in this work to find the most appropriate approach for solving the rhetorical role labeling problem. This section describes the several different methods explored in this work, along with the specific implementation details.

3.1 Baseline Method

Establishing a baseline model for shared tasks is an important step, as it helps track the performance of the more sophisticated approaches that are de-

[2]https://github.com/jaindeepali010/LegalEval-2023

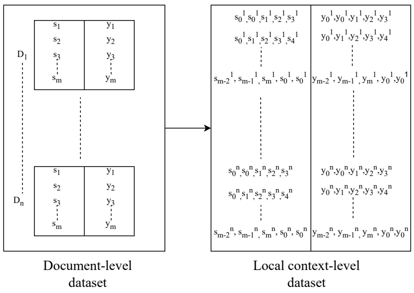

Figure 1: Illustration of Dataset creation process

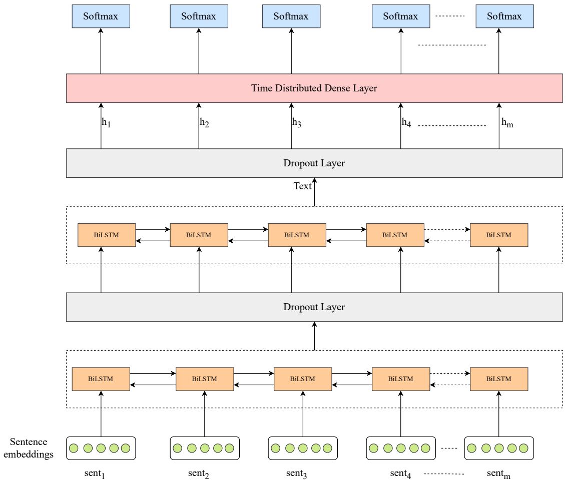

Figure 2: Methodology diagram

veloped in the process of experimental exploration. In this work, our baseline system employs a common technique used in natural language processing (NLP) tasks, where a pre-trained language model is used to encode text data into dense representations, followed by classification layers. More specifically, the Indian Legal BERT (Paul et al., 2022) model is used in this work to encode the sentences into 768-dimensional contextual vectors, which capture the semantic and syntactic information of the text. After encoding the sentences, a multilayer perceptron (MLP) model is employed to classify the encoded sentences into different categories. The MLP model consists of multiple dense layers followed by dropout layers, which help prevent overfitting by randomly dropping out nodes during training. The final softmax layer classifies the encoded sentence into different categories based on the probability distribution. It is important to note here that this approach performs an individual sentence-level classification and does not make use of any context information. This approach is called as the "$Baseline$" approach for all the experimental analysis in the rest of the paper.

3.2 Sentence sequence labeling approach

The primary idea for model development that has been proposed in this work, is a sentence-sequence classification based idea, where a local-context based dataset is built for model training. This dataset is built with a sentence window size of 5, where we hoped to train a sentence-sequence classification model that can learn from it's local context. The reason for restricting the window size to 5 sentences is due to the resource constraints as well as the extremely lengthy nature of legal documents. The dataset building process is pictorially depicted in Fig. 1. Where we generate training samples with a set of 5 sentence embeddings and their corresponding labels. In order to ensure the proper inclusion of each sentence in the dataset regardless of their position in the document, we also perform 0-vector paddings (represented as s_0 in Fig. 1). A BiLSTM based deep learning model was trained on this dataset, the architecture for which is depicted in Fig. 2. During inference we consider two different ways of recovering the sentence level labels: considering the middle sentence in an input sequence as the sentence of interest for recording it's label and considering the last sentence as the sentence of interest. These two approaches are called as the "M_{SS-Mid}" and "M_{SS-End}" respectively.

3.3 Oversampling approach

One important observation from the Task A dataset is that the sentence labels are having a high degree of class-imbalance. Such kind of imbalanced data often causes deep learning models to ignore the minority classes and perform poorly during inference. The level of class imbalance across the dataset is depicted pictorially in Fig. 3, with the normalized frequencies of each of the individual class present in the dataset. In order to deal with this issue one straightforward idea is to employ class-weighted oversampling of minority-class samples. The following is the description of the proposed oversampling technique that we carry out to deal with the class-imbalance problem.

Firstly, we decide to keep only one copy of any 5-sentence sample where at least one of the labels is a dummy label for a 0-vector sentence (s_0). Secondly, for all the other types of samples, we utilize the normalized frequencies of each class to calculate the exact number of times we will oversample them. This calculation is described below:

Let $f_1, f_2, ..., f_{13}$ be the normalized frequencies

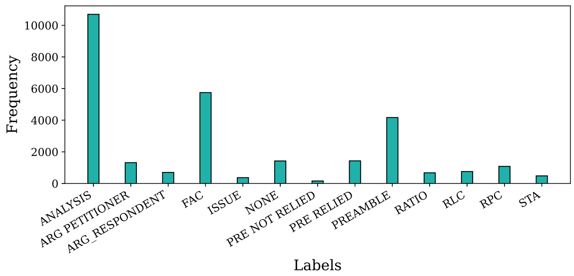

Figure 3: Frequency of class labels in training dataset

Table 1: Classwise Precision scores on dev dataset

Class	$Baseline$	M_{SS-Mid}	M_{SS-End}	$M_{SSO-Mid}$	$M_{SSO-End}$
ANALYSIS	**0.7744**	0.7214	0.7083	0.741	0.7222
ARG PETITIONER	0.3000	**0.3077**	0.2334	0.2457	0.1628
ARG RESPONDENT	0.0526	0.3704	**0.4483**	0.2414	0.2259
FAC	0.6086	0.7578	0.7407	0.7569	**0.7623**
ISSUE	0.7400	**0.8334**	0.8	0.7347	0.6793
NONE	0.8421	0.9603	0.9345	**0.9747**	0.9402
PRE NOT RELIED	0.0	0.0	0.0	**0.25**	0.0
PRE RELIED	0.1901	0.5852	**0.6137**	0.4274	0.4249
PREAMBLE	0.6969	0.842	0.8322	**0.912**	0.903
RATIO	0.0286	0.5637	0.4762	**0.5715**	0.25
RLC	0.1810	0.6334	0.5965	**0.697**	0.6905
RPC	0.7802	**0.9269**	0.8519	0.924	0.8675
STA	0.4286	0.5417	**0.577**	0.5162	0.51516
Weighted Average	0.6231	0.72	0.7033	**0.7255**	0.7005

of class labels for the 13 classes present in the dataset. For the k^{th} sample in the training set, we initially have five labels $[y_1^k, y_2^k, y_3^k, y_4^k, y_5^k]$. Based on these labels we find the following oversampling rate as depicted in equation 1:

$$N_k \quad = \quad \frac{\alpha}{\sum_{i=1}^{5} f_{y_i^k}} \tag{1}$$

where, N_k denotes the number of times the k^{th} sample is to be oversampled, α is a hyperparameter which is chosen experimentally as 5, and the summation term adds up the individual normalized class frequencies of the sentences present in that sample. Such a calculation ensures that the samples with majority class sentences get a smaller amount of oversampling, whereas samples with minority class sentences gets oversampled more number of times.

This oversampling technique gives rise to two new variants of the proposed approach discussed in subsection 3.2, which are referred to as "$M_{SSO-Mid}$" and "$M_{SSO-End}$" in the rest of the paper.

4 Experimental results and discussion

4.1 Experimental setup

The specific architecture used for the $Baseline$ model includes a dense layer of 2048 nodes followed by a dropout layer with 0.6 probability, then another dense layer of 1024 nodes followed by another identical dropout layer, and finally a softmax layer which classifies a given sentence into different categories. An Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.001 is considered. A sparse categorical cross entropy is considered for calculating the loss since the output labels are considered as integer values. For the "M_{SS-Mid}", "M_{SS-End}", "$M_{SSO-Mid}$" and "$M_{SSO-End}$" models, we consider two BiLSTM layers consisting of 128 nodes having a dropout layer with 0.6 probability between them, followed by a time distributed dense layer. The learning rate, optimizer, and loss function is considered the same as the $Baseline$ model. All the models are run for 500 epochs with an early stopping patience of 10.

4.2 Results and discussion

In this section, we present the results of our experiments on the development dataset provided by the organizers along with a detailed discussion on the experimental observations. Specifically, Tables 1, 2, and 3 report the class-wise precision, recall, and F1 scores achieved by our models on the development set. Additionally, we provide a visual representation of the overall weighted F1 scores attained by our various models on the development set through Fig. 4.

Considering the class-wise Precision scores presented in Table 1, we can make certain key observations. The "$M_{SSO-Mid}$" model is able to obtain the best results for five classes in total, however, the performance of this model is not as good on the other class sentences. On the other hand, the "M_{SS-Mid}" model finds best Precision values for three classes with other class results also being quite decent. One key observation is that, the oversampling based method "$M_{SSO-Mid}$" is able to obtain a non-zero Precision for the "PRE NOT RELIED" class, which is a minority class containing the least number of samples in the entire dataset.

The class-wise Recall scores for the different proposed approaches are shown in Table 2. A similar trend as Table 1 can be observed for Recall scores also, where the "M_{SS-Mid}" model achieves improved scores across most of the classes, however the maximum number of best Recall scores are obtained by the oversampling based approach "$M_{SSO-Mid}$". Moreover, the performance of the "$M_{SSO-Mid}$" on the minority classes are better than the non-oversampling based approach.

Table 2: Classwise Recall scores on dev dataset

Class	*Baseline*	M_{SS-Mid}	M_{SS-End}	$M_{SSO-Mid}$	$M_{SSO-End}$
ANALYSIS	0.5844	**0.8446**	0.8242	0.7907	0.7714
ARG PETITIONER	0.1615	0.1143	0.1	**0.2**	0.1
ARG RESPONDENT	**1.0000**	0.2632	0.3422	0.3685	0.3685
FAC	0.6788	0.7552	**0.7828**	0.7621	0.7794
ISSUE	0.6607	0.7	0.64	**0.72**	**0.72**
NONE	0.8511	0.8895	**0.9**	0.8106	0.8264
PRE NOT RELIED	0.0	0.0	0.0	**0.0834**	0.0
PRE RELIED	0.6136	0.3874	0.3803	**0.7043**	0.6972
PREAMBLE	0.7181	**0.9331**	0.9272	0.875	0.878
RATIO	**1.0000**	0.4429	0.1429	0.3429	0.0858
RLC	**0.8077**	0.3276	0.2932	0.1983	0.25
RPC	0.7634	0.8352	0.7583	**0.9341**	0.7913
STA	0.5714	0.4643	0.5358	0.5715	**0.6072**
Weighted Average	0.6474	**0.7363**	0.7245	0.7281	0.7109

Table 3: Classwise F1 scores on dev dataset

Class	*Baseline*	M_{SS-Mid}	M_{SS-End}	$M_{SSO-Mid}$	$M_{SSO-End}$
ANALYSIS	0.6661	**0.7781**	0.7619	0.765	0.746
ARG PETITIONER	0.2100	0.1667	0.14	**0.2205**	0.1239
ARG RESPONDENT	0.1000	0.3077	**0.3881**	0.2917	0.28
FAC	0.6418	0.7565	**0.7612**	0.7595	0.7707
ISSUE	0.6981	**0.7609**	0.7112	0.7273	0.6991
NONE	0.8466	**0.9235**	0.9169	0.8851	0.8796
PRE NOT RELIED	0.0	0.0	0.0	**0.125**	0.0
PRE RELIED	0.2903	0.4662	0.4696	**0.532**	0.528
PREAMBLE	0.7073	0.8852	0.8771	0.893	**0.891**
RATIO	0.0556	**0.496**	0.2198	0.4286	0.1277
RLC	0.2958	**0.4319**	0.3931	0.3088	0.3671
RPC	0.7717	0.8787	0.8024	**0.929**	0.8276
STA	0.4898	0.5	0.5556	0.5424	**0.5574**
Weighted Average	0.5972	**0.7211**	0.7055	0.7195	0.6992

Class-wise F1 scores for the proposed approaches are depicted in Table 3, which demonstrates the overall high performance of the "M_{SS-Mid}" model with five best F1 scores. However, it fails in comparison with the oversampling based approach "$M_{SSO-Mid}$" on the minority classes.

From the overall weighted F1 scores shown in Fig. 4, we can observe that the "M_{SS-Mid}" model achieves the best results and it's oversampling based variant is able to obtain the second best scores.

The findings from the experimental study can be summarized as follows:

- Considering the task of rhetorical role labeling as an individual sentence classification task is not an appropriate approach as it loses a lot of context information. The local-context based sentence sequence labeling approach is able to outperform the single sentence classification based approach in almost all cases.
- Considering the sentence of importance in the middle of the local context is better than to consider it in the end of the context, as "M_{SS-Mid}" model outperforms "M_{SS-End}" model in almost all the scenarios. This is due to the fact that the sentence in the middle has access to it's previous as well next sentence context information, however the sentence at the end only has access to it's previous sentences and their labels.
- Although oversampling based approaches do not outperform the non-oversampling based approaches when the overall performance is considered, they are still quite important, especially for the minority class sentences. Moreover, based on the weighted average scores from Tables 1, 2 and 3 it is quite evident that the oversampling based approach "$M_{SSO-Mid}$" consistently gives impressive precision and recall results.

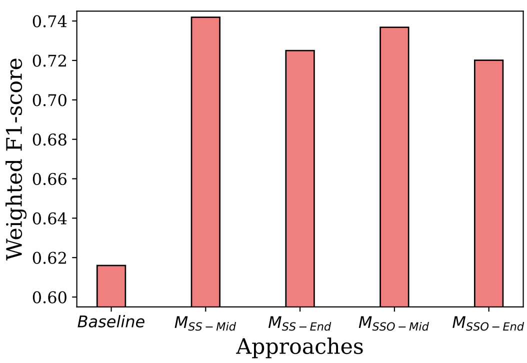

Figure 4: Overall weighted F1 score on dev dataset

5 Conclusion

The organizers of Legal Eval 2023 have introduced a rhetorical Roles Prediction task (Task A) as part of their competition, due to the reputation of Indian case judgments for being lengthy and unstructured. Our team participated in this task and achieved a 71.43% F1 score on the testing data, as reported on the Codalab[3] leaderboard. We also conducted exploratory experiments and discovered that context is a significant factor in accurately identifying the rhetorical roles. Also when it comes to roles that are very rare across such documents, oversampling based model training can actually be quite helpful.

As part of the future work, an ensembling based approach can be explored that combines the predictive power of both oversampling as well as non-oversampling based approaches. Such an approach has the potential to achieve even higher quality rhetorical role labeling.

[3] https://codalab.lisn.upsaclay.fr/competitions/9558

References

Paheli Bhattacharya, Kaustubh Hiware, Subham Rajgaria, Nilay Pochhi, Kripabandhu Ghosh, and Saptarshi Ghosh. 2019a. A comparative study of summarization algorithms applied to legal case judgments. In *European Conference on Information Retrieval*, pages 413–428. Springer.

Paheli Bhattacharya, Shounak Paul, Kripabandhu Ghosh, Saptarshi Ghosh, and Adam Wyner. 2019b. Identification of rhetorical roles of sentences in indian legal judgments. In *Legal Knowledge and Information Systems - JURIX 2019: The Thirty-second Annual Conference, Madrid, Spain, December 11-13, 2019*, volume 322 of *Frontiers in Artificial Intelligence and Applications*, pages 3–12. IOS Press.

Arthur Brack, Anett Hoppe, Pascal Buschermöhle, and Ralph Ewerth. 2021. Sequential sentence classification in research papers using cross-domain multi-task learning. *arXiv e-prints*, pages arXiv–2102.

Ilias Chalkidis, Manos Fergadiotis, Sotiris Kotitsas, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. An empirical study on large-scale multi-label text classification including few and zero-shot labels. *arXiv preprint arXiv:2010.01653*.

Atefeh Farzindar and Guy Lapalme. 2004. Letsum, an automatic legal text summarizing system. *Legal knowledge and information systems, JURIX*, pages 11–18.

Ben Hachey and Claire Grover. 2006. Extractive summarisation of legal texts. *Artificial Intelligence and Law*, 14:305–345.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2020. Fine-tuning textrank for legal document summarization: A bayesian optimization based approach. In *Forum for Information Retrieval Evaluation*, pages 41–48.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2021a. Automatic summarization of legal bills: A comparative analysis of classical extractive approaches. In *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, pages 394–400. IEEE.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2021b. Cawesumm: A contextual and anonymous walk embedding based extractive summarization of legal bills. In *Proceedings of the 18th International Conference on Natural Language Processing (ICON)*, pages 414–422.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2021c. Summarization of indian legal judgement documents via ensembling of contextual embedding based mlp models.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2021d. Summarization of legal documents: Where are we now and the way forward. *Computer Science Review*, 40:100388.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2022. Improving kullback-leibler based legal document summarization using enhanced text representation. In *2022 IEEE Silchar Subsection Conference (SILCON)*, pages 1–5. IEEE.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2023a. Bayesian optimization based score fusion of linguistic approaches for improving legal document summarization. *Knowledge-Based Systems*, page 110336.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2023b. A sentence is known by the company it keeps: Improving legal document summarization using deep clustering. *Artificial Intelligence and Law*, pages 1–36.

Prathamesh Kalamkar, Aman Tiwari, Astha Agarwal, Saurabh Karn, Smita Gupta, Vivek Raghavan, and Ashutosh Modi. 2022. Corpus for automatic structuring of legal documents. *arXiv preprint arXiv:2201.13125*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Vijit Malik, Rishabh Sanjay, Shouvik Kumar Guha, Shubham Kumar Nigam, Angshuman Hazarika, Arnab Bhattacharya, and Ashutosh Modi. 2021a. Semantic segmentation of legal documents via rhetorical roles. *arXiv preprint arXiv:2112.01836*.

Vijit Malik, Rishabh Sanjay, Shubham Kumar Nigam, Kripa Ghosh, Shouvik Kumar Guha, Arnab Bhattacharya, and Ashutosh Modi. 2021b. Ildc for cjpe: Indian legal documents corpus for court judgment prediction and explanation. *arXiv preprint arXiv:2105.13562*.

Ashutosh Modi, Prathamesh Kalamkar, Saurabh Karn, Aman Tiwari, Abhinav Joshi, Sai Kiran Tanikella, Shouvik Guha, Sachin Malhan, and Vivek Raghavan. 2023. SemEval-2023 Task 6: LegalEval: Understanding Legal Texts. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, Toronto, Canada. Association for Computational Linguistics (ACL).

Joel Niklaus, Matthias Stürmer, and Ilias Chalkidis. 2022. An empirical study on cross-x transfer for legal judgment prediction. *arXiv preprint arXiv:2209.12325*.

Shounak Paul, Arpan Mandal, Pawan Goyal, and Saptarshi Ghosh. 2022. Pre-training transformers on indian legal text. *arXiv preprint arXiv:2209.06049*.

Seth Polsley, Pooja Jhunjhunwala, and Ruihong Huang. 2016. Casesummarizer: a system for automated summarization of legal texts. In *Proceedings of COLING 2016, the 26th international conference on Computational Linguistics: System Demonstrations*, pages 258–262.

M Saravanan, Balaraman Ravindran, and S Raman. 2008. Automatic identification of rhetorical roles using conditional random fields for legal document summarization. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*.

MN Satwick Gupta, NL Siva Narayana, V Sai Charan, Kunam Balaram Reddy, Malaya Dutta Borah, and Deepali Jain. 2022. Extractive summarization of indian legal documents. In *Edge Analytics: Select Proceedings of 26th International Conference—ADCOM 2020*, pages 629–638. Springer.

Simone Teufel and Marc Moens. 2002. Summarizing scientific articles: experiments with relevance and rhetorical status. *Computational linguistics*, 28(4):409–445.