YNU-HPCC at SemEval-2023 Task 6: LEGAL-BERT based Hierarchical BiLSTM with CRF for Rhetorical Roles Prediction

Yu Chen, You Zhang, Jin Wang, and **Xuejie Zhang**
School of Information Science and Engineering
Yunnan University
Kunming, China
Contact: chenyv@mail.ynu.edu.cn, yzhang0202@ynu.edu.cn

Abstract

To understand a legal document for real-world applications, SemEval-2023 Task 6 proposes a shared Subtask A, rhetorical roles (RRs) prediction, which requires a system to automatically assign a RR label for each semantical segment in a legal text. In this paper, we propose a LEGAL-BERT based hierarchical BiLSTM model with conditional random field (CRF) for RR prediction, which primarily consists of two parts: word-level and sentence-level encoders. The word-level encoder first adopts a legal-domain pre-trained language model, LEGAL-BERT, initially word-embedding words in each sentence in a document and a word-level BiLSTM further encoding such sentence representation. The sentence-level encoder then uses an attentive pooling method for sentence embedding and a sentence-level BiLSTM for document modeling. Finally, a CRF is utilized to predict RRs for each sentence. The officially released results show that our method outperformed the baseline systems. Our team won 7th rank out of 27 participants in Subtask A.

1 Introduction

In populous countries, the number of pending legal cases has been rising exponentially. For instance, in India, according to the National Judicial Data Grid[1], as of 04 July 2022, the Supreme Court of India had approximately 6 million cases pending, and with the addition of other local courts, there will be far more than 6 million cases pending. It is urgent to require an automatic legal system, which helps practitioners extract accurate and valid information from legal documents, for efficient legal processing. To this end, SemEval-2023 proposes a shared Task 6 (Modi et al., 2023), LegalEval, for understanding legal texts, which mainly comprises of three subtasks including rhetorical roles (RRs) prediction, legal named entity recognition (L-NER), and court judgment prediction with explanation (CJPE), respectively.

- Subtask A RR Prediction. Structuring unstructured legal documents into semantically coherent units.
- Subtask B L-NER. Identifying relevant entities in a legal document.
- Subtask C CJPE. Predicting the outcome of a case along with an explanation.

Focusing on the RR prediction task, it requires a system to structure unstructured legal documents into semantical segments that are aligned with RR labels such as preamble, fact, ratio, etc. Such segmentations as fundamental building blocks are crucial for many legal artificial intelligence (AI) applications, e.g., judgment summarizing, judgment outcome prediction, and precedent search. Due to the shared task having preprocessed legal documents by splitting them into several semantical sentences as coherent units, RR prediction in Subtask A could be regarded as a sentence-wise sequence labeling task that predicts a RR label for each sentence in a legal document.

With the increasing growth of deep learning (DL) and advanced pre-trained language models (PLMs) (Devlin et al., 2019), various natural language processing (NLP) tasks have been effectively addressed, such as sentiment analysis (Wang et al., 2018; Zhang et al., 2021b,a), named entity recognition (Zhou and Su, 2002), and text classification (Soni et al., 2022; Cho et al., 2014). For instance, SciBERT-HSLN (Brack et al., 2021) introduced a sequential sentence classification method for RR predictions. To further consider the problem of low shift probability in sentence labels, several works adopted label shift prediction as an auxiliary task for accurate RR prediction (Kalamkar et al., 2022). Although general PLMs, e.g., BERT (Devlin et al., 2019), RoBERT (Liu et al., 2019), and AlBERT

[1] https://njdg.ecourts.gov.in/hcnjdgnew/?p=main/pend_dashboard

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 2075–2081

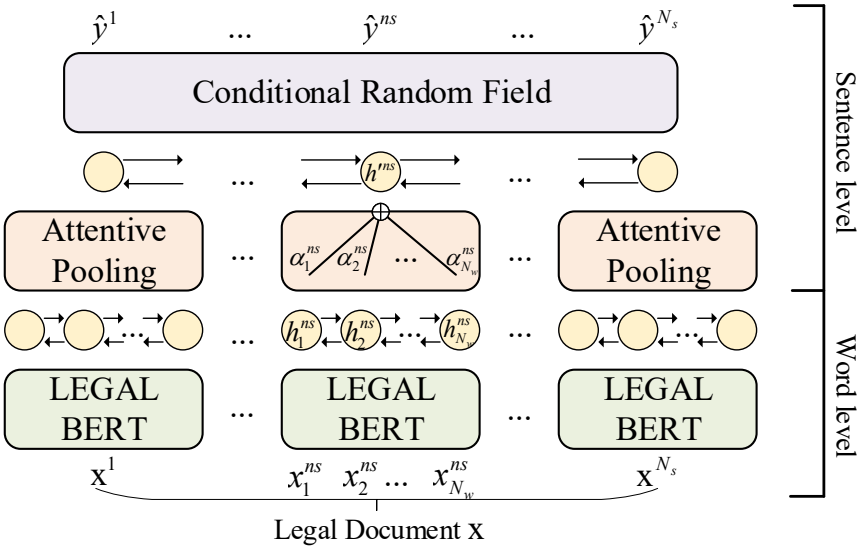

Figure 1: The overview of the proposed method.

(Lan et al., 2019), have shown a robust ability to effectively model most natural language texts and be deployed in real-world applications, they are difficult to apply directly for RR prediction tasks. The main reason is that legal texts differ from general texts used for existing model training. Most of the legal texts are much longer than the input (512 tokens) of most pre-trained models. Moreover, legal texts contain a lot of specialized terminology, and pre-trained models that are not trained on the legal corpus may not work particularly well.

To address the above problem, we proposed LEGAL-BERT-based (Zheng et al., 2021) hierarchical BiLSTM (bidirectional long short-term memory) with conditional random field (CRF) (Bhattacharya et al., 2023), denoted as HLBERT-CRF, for RR prediction. LEGAL-BERT is a legal-specific PLM that further trains the BERT model on legal domain data. HLBERT-CRF primarily consists of two parts: word-level and sentence-level encoder. Word-level encoder based on LEGAL-BERT is first proposed to encode sentence representations via BiLSTM. Sentence representations are then integrated via attention pooling and further encoded via BiLSTM at the sentence level, resulting in a document representation that contains a set of sentence embeddings. Finally, a CRF-based classifier predicts probabilities over various RR labels. Moreover, we found that using a hard-voting strategy gained more performance. A detailed experiment is conducted on Subtask A of SemEval-2023 shared Task 6, revealing our team won the 7th rank out of 27 participant teams.

The remainder of the paper is structured as follows. Section 2 provides a detailed description of the proposed method. Extensive experiments are conducted and analyzed in Section 3, and conclusions are drawn in Section 4.

2 System Description

This section mainly describes the proposed HLBERT-CRF model for RR predictions. As shown in Fig. 1, HLBERT-CRF primarily consists of two parts, including word-level and sentence-level encoders. The word-level encoder is intented to generate sentence representations; the sentence-level encoder aims to generate document representation containing sentence embeddings that are utilized for predicting sentence-wise RR labels by the CRF module. Before introducing the proposed model in detail, we first formulate the RR prediction tasks.

2.1 RR Prediction Task

Given a legal document $\mathbf{x}$, which is semantically segmented into N_s sentences $\mathbf{x} = [\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{N_s}]$. Each sentence $\mathbf{x}^{ns} = [x_1^{ns}, x_2^{ns}, \ldots, x_{N_w}^{ns}]$, $ns \in [1 : N_s]$ contains N_w tokens split by special tokenizers, such as word-piece tokenizer (Devlin et al., 2019). The RR predictor tasks regarded as sentence-wise sequence labeling tasks require a system $f_\theta(\hat{\mathbf{y}}|\mathbf{x})$ to predict the ground-truth sentence labels $\mathbf{y} \in \mathbb{R}^{N_s}$ (Ma et al., 2021), where θ is the whole parameters in the system and each dimension in $\mathbf{y}$ is the ground-truth RR label index in the RR label list with a

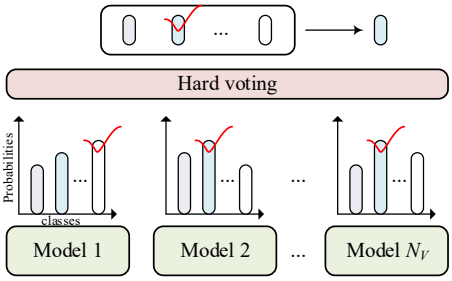

Figure 2: A paradigm of hard-voting strategy.

vocabulary size of C.

2.2 Word-level Encoder

The word-level encoder is built from LEGAL-BERT and word-level BiLSTM. To tokenize each sentence into sequential tokens $\mathbf{x}^{ns}$, we adopt a BERT tokenizer compatible with LEGAL-BERT PLM.

LEGAL-BERT: LEGAL-BERT will first be used as a word embedding tool to project discrete tokens into a high-dimensional real-valued space, for nsth sentence:

$$\mathbf{w}^{ns} = \text{LEGAL-BERT}(\mathrm{x}^{ns}) \in \mathbb{R}^{N_w \times dw}, \quad (1)$$

where dw denote the dimensionality of word embedding.

Due to LEGAL-BERT could feasibly adapt legal domain datasets and generate contextual word embeddings, it facilitates efficient modeling of sentence and document representations in legal texts by hierarchical BiLSTM.

BiLSTM: LSTM is a sequential model, which employs three gate units to address the gradient vanishing problem occurring in traditional recurrent neural networks (RNNs). Given a sentence representation $\mathbf{w}^{ns} = [w_1^{ns}, w_1^{ns}, ..., w_{N_w}^{ns}]$ as the input of an LSTM layer, LSTM cell calculate tth ($t \in [1 : N_w]$) word representation h_t^{ns} as,

$$\begin{aligned}
\mathrm{i}_t &= \sigma(\text{Linear}_i([h_{t-1}^{ns}, w_{t-1}^{ns}])) \\
\mathrm{f}_t &= \sigma(\text{Linear}_f([h_{t-1}^{ns}, w_{t-1}^{ns}])) \\
\mathrm{o}_t &= \sigma(\text{Linear}_o([h_{t-1}^{ns}, w_{t-1}^{ns}])) \\
\tilde{C}_t &= \tanh(\text{Linear}_c([h_{t-1}^{ns}, w_{t-1}^{ns}])) \\
C_t &= \mathrm{f}_t \odot C_{t-1} + \mathrm{i}_t \odot \tilde{C}_t \\
h_t^{ns} &= \mathrm{o}_t \odot \tanh(C_t)
\end{aligned}, \quad (2)$$

where $\text{Linear}_m(\cdot)$ presents the one-layer fully-connected layer, $m \in \{i, f, o, c\}$ stand for input gate, forget gate, output gate, and cell state, respectively; $[\cdot;\cdot]$ is the concatenation manner and $\odot$ is the element-wise multiplication.

LSTM generally model sequence forward, i.e., left-right, lacking further contextual features for sentence representation. Therefore, we also introduce a backward LSTM to model sentence representation from right to left (Lample et al., 2016). Thus, BiLSTM at word level modeling each sentence is simply formulated as,

$$\begin{aligned}
\mathbf{h}^{ns} &= \text{BiLSTM}(\mathbf{w}^{ns}) \in \mathbb{R}^{N_w \times dh} \\
&= [\overrightarrow{\text{LSTM}}(\mathbf{w}^{ns}); \overleftarrow{\text{LSTM}}(\mathbf{w}^{ns})]
\end{aligned}, \quad (3)$$

where dh is the output feature dimension and $\mathbf{h}^{ns} = [h_1^{ns}, h_2^{ns}, ..., h_{N_w}^{ns}]$ is the nsth sentence representation of the output of the word-level encoder.

2.3 Sentence-level Encoder

Sentence-level encoder model sentence representation $\mathbf{h} = [\mathbf{h}^1, \mathbf{h}^2, \ldots, \mathbf{h}^{N_s}] \in \mathbb{R}^{N_s \times N_w \times dh}$ to capture document representation $\mathbf{h}' = h'^1, h'^2, \ldots, h'^{N_s}] \in \mathbb{R}^{N_s \times dh'}$, which mainly comprises of attentive pooling, sentence-level BiLSTM, and CRF-based sequential label classifier.

Attentive pooling: To integrate a sentence representation with sequential tokens into a semantic vector, attentive pooling is introduced. In terms of the nsth sentence representation $\mathbf{h}^{ns} = [h_1^{ns}, h_2^{ns}, ..., h_{N_w}^{ns}]$, its integral representation is denoted as h^{ns} and formulated as,

$$\begin{aligned}
a_{nw} &= \text{score}(h_{nw}^{ns}) \\
&= v^{\mathrm{T}} \tanh(\text{Linear}_{att}(h_{nw}^{ns})) \in \mathbb{R}^1 \\
\alpha_{nw} &= \frac{\exp(a_{nw})}{\sum_i^{Nw} \exp(\text{score}(h_i^{ns}))} \in \mathbb{R}^1 \\
h^{ns} &= \sum_i^{Nw} h_i^{ns} \cdot \alpha_i \in \mathbb{R}^{dh}
\end{aligned}, \quad (4)$$

where v and $\text{Linear}_{att}(\cdot)$ are a trainable vector parameter and a fully-connected layer in attentive pooling, respectively. The nw represents the nwth token in nsth sentences.

Sentence-level BiLSTM: To further model long-range dependency among sentences, we use another BiLSTM applied to integral sentence representations in a document. Based on integral sentence representations $\hat{\mathbf{h}} = [h^1, h^2, ..., h^{N_s}] \in \mathbb{R}^{N_s \times dh}$, the sentence-level BiLSTM calculates

a document representation with sentence embeddings by

$$\mathbf{h}' = \text{BiLSTM}(\hat{\mathbf{h}}) \in \mathbb{R}^{N_s \times dh'}. \quad (5)$$

CRF-based classifier: Following the previous work for sequence labeling tasks (Bhattacharya et al., 2023), we employ a CRF-based classifier to predict the RR label for each sentence embedding. CRF-based classifier considers not only the current sentence representation but also the relatedness among RR labels for RR prediction, formulated as,

$$\hat{y} = \text{CRF-Classifier}(\mathbf{h}'), \quad (6)$$

where $\hat{\mathbf{y}} \in \mathbb{R}^{N_s}$ is predicted RR labels for each sentence.

2.4 Hard Voting Strategy

To further leverage the performance of RR predictions in Subtask A, we use a hard-voting strategy to assemble multiple HLBERT-CRF models for an ultimate inference. As shown in Fig. 2, given N_v HLBERT-CRF models that are initialized with different random seeds and individually trained on the same training datasets, the final sequential label is selected according to the RR labels with the maximum votes recommended by different models in the statistic (Farooqi et al., 2021).

3 Experimental Results

In this section, extensive experiments were conducted and analyzed.

3.1 Datasets

RR prediction tasks provided a RR corpus, which is collected from Indian Court websites and mixed of Supreme Court judgments, High Courts judgments, and district court judgments. The corpus is split into train, dev, and test datasets (Malik et al., 2021). Only train and dev datasets can be available for any participant and used for system optimization. while test datasets used to evaluate the performance of submitted models are peculiar to task organization. A detailed statistic of the corpus is listed in Table 1.

Each entry contains three parts: ID, annotations, and data. ID is the unique identifier of the document; data is the actual judgment; and annotations include the sentence ID, sentence text, sentence position in the judgment text (marked by start and end), and label of the sentence. It should be noted that this is a multi-class classification task where each sentence has only one label. There are 13 kinds of labels such as preamble, fact, ratio, arguments, etc.

Dataset	Docs	Sentences	Tokens	Avg.
Train	247	28,986	938k	3,797
Dev	30	2,879	88k	2,947
Test	77†	8,450†	261k†	7403†

Table 1: The detailed statistics of the RR corpus. Avg. denotes the average tokens per document. † represents figures referred to Malik et al. (2021) and corresponding data not available for participant teams.

3.2 Evaluation Metrics

To evaluate the performance of participant systems, the organizer provides weighted micro-F_1 score as the official metric.

- F_1 score

$$\begin{aligned} \text{Precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}} \\ \text{Recall} &= \frac{\text{TP}}{\text{TP} + \text{FN}} \\ \text{F}_1 &= \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \cdot 2 \end{aligned}, \quad (7)$$

  where TP, FP, and FN represent the number of predicted true-positive, false-positive, and false-negative samples, respectively, given a certain RR label as a positive label and others as negative labels.

- Weighted F_1 score

$$\text{weighted-F}_1 = \sum_{c}^{C} \frac{\text{E}_c}{\text{E}} \text{F}_{1,c}, \quad (8)$$

  where C is the number of RR label types; E_c, $F_{1,c}$, and E denote the number of samples with respect to label c, F_1 score as label c being positive label, and the total number of samples, respectively.

3.3 Implementation Details

Hyper-parameters. For token splitting, the `BERT-base-uncased` tokenizer is used. To build legal documents as hierarchical structures, we chose the maximum number of sentences in each document of a batch of samples as the fixed document length (N_s). For documents less than such

Hyper-parameter	Values
Learning rate	3e-5
Dropout	0.5
Batch size	32
Max epoch	30
Early stopping	5
Word-level BiLSTM hidden state (dh)	768
Sentence-level BiLSTM hidden states (dh')	768
Maximum gradient	1.0
Learning rate epoch decay	0.9
Voting members	20

Table 2: Hyper-parameters in our model.

document length, zeroed sentences are padded at the end of them. Regarding sentence length (N_w), we set it to 128. Then, sentences less or longer than 128 tokens will be zeroed-token padded or truncated. LEGAL-BERT is initialized from well pre-trained checkpoint[2]. The dimensionalities of word embedding (d_w), sentence representation (d_h) and document representation (dh') are identical and set to 768. In terms of optimization, we use Adam as the optimizer and gradient clip strategy for avoid gradient explosion. More detailed hyper-parameter settings can be found in Table 2.

Note that, all parameters are selected in grid search method, monitored on the best Dev performance. The code of this paper is availabled at https://github.com/cy330874054/2023_task6_RR_predition/

Baselines: To investigate the effect of the proposed method, we introduce several baseline models. Initially, we replace the LEGAL-BERT (Zheng et al., 2021) with other PLMs, such as XLNet (Yang et al., 2019), RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019), and BERT (Devlin et al., 2019). Next, we altered hierarchical structures as baselines to investigate effect of the hierarchical BiLSTM.

3.4 Result and Analysis

Comparative results are reported in Table 3, which is categorized into two groups.

The first group comprises hierarchical BiLSTM-CRF models with different PLMs as word embeddings. It can be observed that, with varying embeddings, RR prediction models perform differently. The main reason is that PLMs may learn different specific knowledge during pre-training phases, and their diverse supervision objectives make the learnt information to different extents. The proposed method that utilizes LEGAL-BERT achieves the best results, because LEGAL-BERT is pre-trained in a large legal corpus and easy to facilitate legal-specific downstream tasks, i.e., legal RR prediction.

In the second group, we conduct several ablation studies to investigate the effect of hierarchical structure. First, we replace the word and sentence-level BiLSTM with Transformer layers as HLBERT-CRF w/ Transformer structure, which reduces weighted-F_1 scores. This scenario reveals that BiLSTM is more suitable for building hierarchical structures than Transformer layers. Moreover, with the drops of word or sentence or both-level BiLSTM, corresponding performance is simultaneously degraded, further demonstrating the effect of BiLSTM-based hierarchical structure for RR predictions.

Finally, the proposed HLBERT-CRF model won the 7th rank out of 27 participant teams in SemEval-2023 shared Task 6 (Subtask A) with a weighted-F_1 score of 0.8146. Table 4 reported feedback test performance in two submissions. The proposed model introducing a hard-voting ensemble strategy slightly outperforms the model without hard-voting.

Model	weighted-F_1
XLNet‡	0.5722
RoBERTa‡	0.6330
ALBERT‡	0.6863
BERT‡	0.7918
HLBERT-CRF	**0.8079**
HLBERT-CRF	
$w/$ Transformer structure	0.6276
w/o word-level BiLSTM	0.7803
w/o sentence-level BiLSTM	0.6248
w/o both-level BiLSTM	0.5633

Table 3: Comparative weight-F_1 score on Dev dataset. PLMs‡ mean they substitute for LEGAL-BERT in HLBERT-CRF.

Model	weighted-F_1
HLBERT-CRF	0.8140
HLBERT-CRF $w/$ hard vote	0.8146

Table 4: Test weight-F_1 scores of $w/$ and w/o hard-voting.

[2] https://huggingface.co/docs/transformers

In summary, the priority of the proposed model is three-fold: 1) introducing a legal-specific PLM to initialize word embeddings; 2) regarding RR prediction as a sentence-wise sequence labeling task and employing a hierarchical BiLSTM with CRF as the backbone; 3) adopting hard-voting ensemble strategy for performance improvement.

4 Conclusions

In this paper, we proposed an HLBERT-CRF model for RR prediction. HLBERT-CRF utilizes hierarchical BiLSTM with CRF structure the backbone and introduces legal-specific PLM, LEGAL-BERT, for knowledge transferring. As a result, HLBERT-CRF with a hard-voting strategy won the 7th in SemEval-2023 shared Task 6 (Subtask A).

In the future, we will explore more complex legal text-based NLP tasks by devising more efficient structures and explainable modules.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant Nos. 61966038 and 62266051, and the Yunnan Postdoctoral Science Foundation under Grant Nos. C615300504048. The authors would like to thank the anonymous reviewers for their constructive comments.

References

Paheli Bhattacharya, Shounak Paul, Kripabandhu Ghosh, Saptarshi Ghosh, and Adam Wyner. 2023. Deeprhole: deep learning for rhetorical role labeling of sentences in legal case documents. *Artificial Intelligence and Law*, 31(1):53–90.

Arthur Brack, Anett Hoppe, Pascal Buschermöhle, and Ralph Ewerth. 2021. Sequential sentence classification in research papers using cross-domain multi-task learning. *arXiv eprint arXiv:2102.06008v2*.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference (EMNLP-2014)*, pages 1724–1734.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova Google, and A I Language. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-2019)*, pages 4171–4186.

Zaki Mustafa Farooqi, Sreyan Ghosh, and Rajiv Ratn Shah. 2021. Leveraging Transformers for Hate Speech Detection in Conversational Code-Mixed Tweets. *arXiv eprint arXiv:2112.09986*.

Prathamesh Kalamkar, Aman Tiwari, Astha Agarwal, Saurabh Karn, Smita Gupta, Vivek Raghavan, and Ashutosh Modi. 2022. Corpus for automatic structuring of legal documents. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC-2022)*, pages 4420–4429, Marseille, France. European Language Resources Association.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-2016)*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *arXiv preprint arXiv:1909.11942*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*.

Xinge Ma, Jin Wang, and Xuejie Zhang. 2021. YNU-HPCC at SemEval-2021 task 11: Using a BERT model to extract contributions from NLP scholarly articles. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 478–484.

Vijit Malik, Rishabh Sanjay, Shouvik Kumar Guha, Angshuman Hazarika, Shubham Nigam, Arnab Bhattacharya, and Ashutosh Modi. 2021. Semantic Segmentation of Legal Documents via Rhetorical Roles. *arXiv eprint arXiv:2112.01836*.

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).

Sanskar Soni, Satyendra Singh Chouhan, and Santosh Singh Rathore. 2022. Textconvonet: a convolutional neural network based architecture for text classification. *Applied Intelligence*.

Jin Wang, Bo Peng, and Xuejie Zhang. 2018. Using a stacked residual lstm model for sentiment intensity prediction. *Neurocomputing*, 322:93–101.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *Advances in Neural Information Processing Systems (NeurIPS-2019)*, volume 32, pages 5753–5763. Curran Associates, Inc.

You Zhang, Jin Wang, Liang-Chih Yu, and Xuejie Zhang. 2021a. Ma-bert: learning representation by incorporating multi-attribute knowledge in transformers. In *Findings of the Association for Computational Linguistics (ACL-IJCNLP-2021)*, pages 2338–2343.

You Zhang, Jin Wang, and Xuejie Zhang. 2021b. Conciseness is better: Recurrent attention lstm model for document-level sentiment analysis. *Neurocomputing*, 462:101–112.

Lucia Zheng, Neel Guha, Brandon R. Anderson, Peter Henderson, and Daniel E. Ho. 2021. When does pretraining help? assessing self-supervised learning for law and the casehold dataset of 53,000+ legal holdings. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law (ICAIL-2021)*, page 159–168, New York, NY, USA. Association for Computing Machinery.

GuoDong Zhou and Jian Su. 2002. Named entity recognition using an hmm-based chunk tagger. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL-2002)*, page 473–480, USA. Association for Computational Linguistics.