LRL_NC at SemEval-2023 Task 6: Sequential Sentence Classification for Legal Documents using Topic Modeling Features

Kushagri Tandon, Niladri Chatterjee
Department of Mathematics
Indian Institute of Technology Delhi
Hauz Khas, Delhi-110016, India
{kushagri.tandon,niladri.chatterjee}@maths.iitd.ac.in

Abstract

Natural Language Processing techniques can be leveraged to process legal proceedings for various downstream applications, such as summarization of a given judgement, prediction of the judgement for a given legal case, precedent search, among others. These applications will benefit from legal judgement documents already segmented into topically coherent units. The current task, namely, Rhetorical Role Prediction, aims at categorising each sentence in the sequence of sentences in a judgement document into different labels. The system proposed in this work combines topic modeling and RoBERTa to encode sentences in each document. A BiLSTM layer has been utilised to get contextualised sentence representations. The Rhetorical Role predictions for each sentence in each document are generated by a final CRF layer of the proposed neuro-computing system. This system secured the rank 12 in the official task ranking, achieving the micro-F1 score 0.7980. The code for the proposed systems has been made available at https://github.com/KushagriT/SemEval23_LegalEval_TeamLRL_NC

1 Introduction

Legal domain hosts a wide range of text documents corresponding to different stages of the legal proceedings. Especially in populous countries, such as India, which has a huge and branched judicial system, automation of certain steps in the judicial pipeline can be beneficial to the overall working of the system. Legal documents are generally long and unstructured, and typically consist of jargons specific to the legal domain. Due to presence of various subdomains, such as criminal law, civil law, income-tax law, among others, the systems developed on one domain may not generalise well to other domains. Additionally, the legal domain suffers from lack of availability of annotated corpora (Kalamkar et al., 2022). All the above observations make the task truly challenging.

Natural Language Processing techniques can be leveraged to process such corpora for various downstream applications, such as judgement summarising, judgement outcome prediction, precedent search, among others. These applications will benefit from legal judgement documents that are already segmented into topically coherent units. These coherent units are called Rhetorical Roles. This task of semantic segmentation of a judgement document, into Rhetorical Roles, aids in the overall processing of legal documents.

The aim of the current task (Modi et al., 2023) is to categorise sentences in a legal case proceeding (judgement document), to different labels. Thus the task effectively boils down to Rhetorical Role Prediction. The dataset (Kalamkar et al., 2022; Malik et al., 2021a) consists of 13 Rhetoric Role labels given in Table 1.

This paper proposes a system for Rhetorical Role prediction for legal judgement documents. The proposed approach uses a hierarchical sequence encoder consisting of basic sentence representation and a contextualised sentence representation followed by a CRF (Lafferty et al., 2001) layer for final label prediction. Experiments were conducted with several variations of the proposed scheme, and the best performing one is discussed in detail in this paper.

The novelty of the proposed approach lies in the use of topic modeling augmented with transformer-based (RoBERTa) sentence embeddings to generate sentence representations. This combination of RoBERTa with topic modeling to generate sentence representations has been found to be more effective than just using transformer-based embedding.

The paper is organised as follows. The related past works are discussed in Section 2. Section 3 and Section 4 discuss the details of the proposed system and the experimental setup, respectively. The results from the systems are given in Section 5. The paper is concluded in Section 6.

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 143–149

Rhetoric Role	Label
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	NONE

Table 1: Rhetoric Role Labels

2 Related Works

Rhetorical Role Prediction aims at classifying each sentence in a given sequence of sentences into one of the 13 classes as mentioned in Table 1. This task is more challenging than a traditional classification task, where each instance is considered independent of other instances. This is not a valid assumption for sequential sentence classification, which makes the task more challenging.

Most of the existing works use Conditional Random Fields (CRFs) (Lafferty et al., 2001) as an important component in the architecture for sequential sentence labelling. These works primarily use hierarchical sequence encoders to contextualise sentence representations, followed by a CRF for the sequential classification in each document since they incorporate dependencies between subsequent labels. In particular, (Bhattacharya et al., 2019) and (Malik et al., 2021a) use BiLSTM-CRF as models for Rhetorical Role Prediction.

Jin and Szolovits (2018) propose a hierarchical neural network model for sequential sentence classification task, which we call a hierarchical sequential labeling network (HSLN). This consists of an RNN or CNN layer and a BiLSTM layer to get sentence representation and contextualised sentence representation, respectively. This is followed by a single hidden layer feedforward network to transform the sentence representation to the probability vector. The final layer is CRF to jointly optimise the predicted label sequence.

In recent years there has been a surge in pre-trained transformer-based models for most of the Natural Language Processing tasks. In particular, for the task of sequential sentence classification, Cohan et al. (2019) use pre-trained language models, such as BERT (Devlin et al., 2019) to capture contextual dependencies without the need for neither hierarchical encoding nor a CRF. Their approach achieved state-of-the-art results on four datasets exhibited.

3 System Overview

In the first step the system extracts individual sentence representations that are not contextualised with respect to the whole document. This step uses RoBERTa (Liu et al., 2019) for generating text embeddings. The RoBERTa model has been fine-tuned for Masked Language Modeling (MLM) task on the training and development subsets of the related ILDC (Indian Legal Documents Corpus) dataset (Malik et al., 2021b). Sentence representation is the CLS token embedding for that sentence. This fine-tuned RoBERTa Model is applied to each sentence of each document to get corresponding sentence embeddings which are the CLS embeddings from the RoBERTa model.

The overall scheme may be explained as follows: Suppose each document D is represented by a $S_D \times L$ matrix, where S_D is the number of sentences in the document and L is the maximum length of all sentences in the document, where each sentence with less number of words than L are padded with a padding token. This step generates an embedding for each sentence in the document, and thereby converts a document from $S_D \times L$ to $S_D \times RoBERTa\_Embedding\_Size$.

The existing sentence encoding from RoBERTa

model accounts for the information from all the words of the sentence and how the words interact. But, to account for topical composition of each sentence with respect to the information learned from the entire corpus, the features are augmented using a topic modeling approach. The idea is to train a nonparametric topic model, namely, Hierarchical Dirichlet Process (HDP) (Teh et al., 2006) on the collection of training documents, which learns the number of topics K from the data itself. Application of HDP on a sentence produces a K-dimensional vector which represents the topical composition of that sentence. The per-document per-sentence preliminary representation obtained from RoBERTa model in the previous step is concatenated with HDP topic proportions per sentence, to get a combined sentence representation. This results in a $S \times (RoBERTa\_Embedding\_Size + K)$ matrix of feature vectors corresponding to each document.

Using these combined sentence embeddings as features a document level BiLSTM of hidden state dimension $Hidden\_Size\_Document$ is applied to each document to get hidden states corresponding to each sentence. This results in features of dimension $S \times (2 \times Hidden\_Size\_Document)$ for each document, consisting of concatenated hidden states from backward and forward LSTM. These hidden states are input to a linear neural network layer. The output is pseudo-prediction vectors corresponding to the tag space of size $Target\_Size$ which is the number of labels in the target space, resulting in predictions of dimension $S \times (Target\_Size)$.

Up to this point the model has taken into account features of neighbouring sentences for predicting the Rhetorical Role (Table 1) for a sentence. The final layer is the CRF layer which takes emissions, masks and actual labels as input while training. The emissions are taken as the pseudo-predictions generated in the previous step. The masks consist of zeros corresponding to the sentences which solely consist of padding tokens. The CRF layer takes into account document-level label information to generate prediction for the current sentence. The loss for training this model is negative loglikelihood from the CRF layer. Viterbi algorithm (Viterbi, 1967) is used for the prediction for a new document.

Figure 1 provides the diagram for the proposed system. Two variations of the proposed approach were also submitted to the task.

- In one variation the HDP features were discarded.
- In the other variation, the nonparametric HDP model was substituted with parametric Topic Model, Correlated Topic Model (CTM) (Lafferty and Blei, 2005).

Table 4 refers to these variations as Proposed - HDP and Proposed - HDP + CTM, respectively. The results for these variations are discussed in Section 5.

4 Experimental Setup

The experiments were carried on Google Colaboratory in Python 3.8.10 with Nvidia Tesla P100 GPU. PyTorch (Paszke et al., 2019), Huggingface Transformers (Wolf et al., 2020), and pytorch-crf[1] are used as the key frameworks for the experiments.

For fine-tuning RoBERTa for MLM task the following settings were followed.

- The instances in the ILDC dataset were prepared using RobertaTokenizerFast from Huggingface Transformers to tokenize the text by masking tokens in the input with probability 0.15. The instances were padded to the maximum sequence length in the batch. Truncation was done at a maximum sequence length of 256. RoBERTa special tokens were added.
- For fine-tuning each sentence from each document is considered as an individual instance.
- The roberta-base model was fine-tuned. The model was implemented using RobertaForMaskedLM from Huggingface Transformers.
- No preprocessing is applied to the ILDC dataset, except the removal of the newline character (‘\ n’) from the text.
- For fine-tuning, Trainer functionality of Huggingface Transformers was used, with batch size 8, for 10 training epochs[2].
- Learning rate of 5e-5 is used with a linear learning rate scheduler.

To prepare the text for training the main model, the newline character, ellipsis and comma are replaced by whitespace. The text is tokenized using

[1] https://pytorch-crf.readthedocs.io/en/stable/

[2] The model checkpoint at 20000 training steps is used.

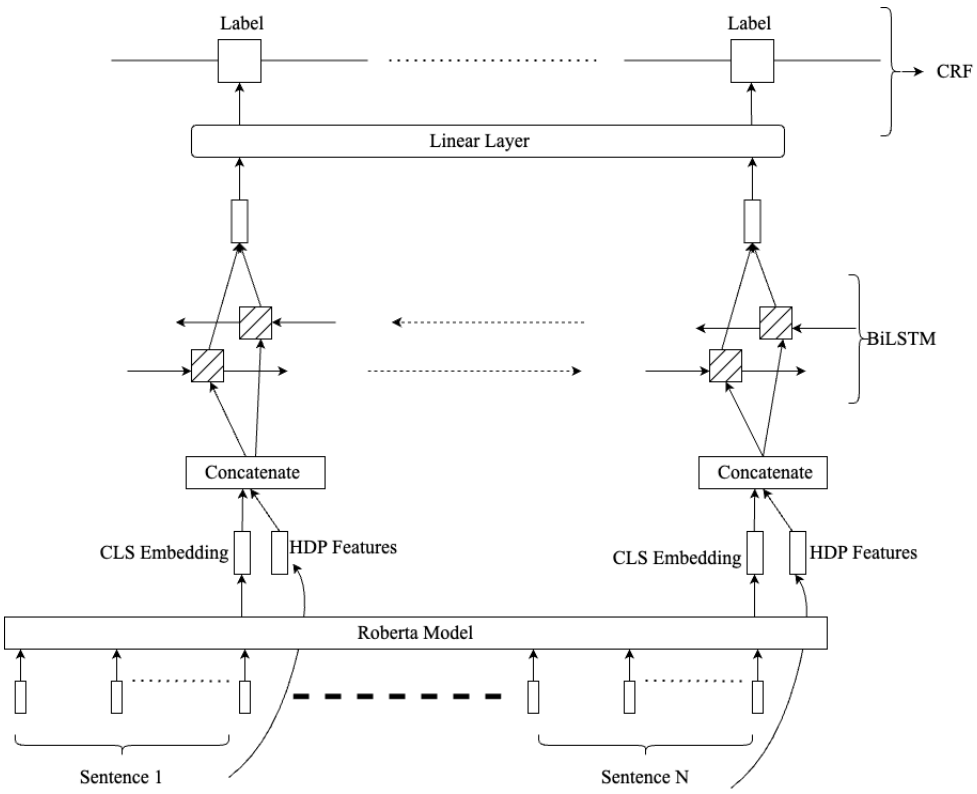

Figure 1: System for Rhetoric Role Prediction

RobertaTokenizerFast tokenizer pretrained from roberta-base model. For each sentence, padding and truncation are enabled, with padding to maximum sequence length in the batch and truncation at maximum sequence length of 256. RoBERTa special tokens have been added.

To the number of tags, two additional tags, namely, <START> and <STOP> are added. For batching, documents are sorted according to the number of sentences in the document. Top B documents are selected and are represented as $S_1 \times L_1, S_2 \times L_2, \cdots, S_B \times L_B$, where B is the chosen batch size. The maximum number of sentences in the batch are $S = max(S_1, \cdots, S_B)$ and maximum length of sequence in the batch is $L = max(L1, \cdots, L_B)$. Each sentence in each document in the batch is padded with padding token (<pad>) id. In case of RoBERTa, this <pad> token is assigned the id 0. Thus the documents are now of the sizes, $S_1 \times L, \cdots, S_B \times L$. Each document is padded to the size $S \times L$, by adding sentences consisting only of <pad> sequences, with corresponding labels as <STOP>. This prepares a batch of size $B \times S \times L$ and corresponding labels $B \times S$. Corresponding mask vectors of size $B \times S \times L$ are prepared, with entries 0 or 1, with 0 indicating presence of the <pad> token and 1 indicating the absence of the <pad> token.

The model settings are given in Table 2. Adam (Kingma and Ba, 2014) optimizer has been used for training the parameters.

Experiment Setting	Value
RoBERTa_Embedding_Size	768
Hidden_Size_Document	50
Batch Size	2
Number of Epochs	25
Optimizer	Adam
Learning rate	0.0001

Table 2: Model Settings

While training the model, all the parameters of the RoBERTa layer are frozen, except for parameters of the pooler layer and the last encoder layer. To implement CRF, pytorch-crf package is used. The HDP model is trained using tomotopy package[3] in Python. The text is preprocessed by removing the newline character, ellipsis, and comma. Additionally preprocessing[4] functionality of gensim (Řehůřek and Sojka, 2010) package is used to clean the data. A bigram model is trained on the

[3] https://bab2min.github.io/tomotopy/v/en/
[4] simple_prepocess

Model	Micro F1
RoBERTa + HDP + BiLSTM + CRF	**0.7930**
RoBERTa + BiLSTM + CRF	0.7697
RoBERTa + CTM + BiLSTM + CRF	0.7864
RoBERTa(no FT) + HDP + BiLSTM + CRF	0.7506
RoBERTa(no FT) + BiLSTM + CRF	0.7860
RoBERTa(no FT) + CTM + BiLSTM + CRF	0.7860
BERT(no FT) + HDP + BiLSTM + CRF	0.7867
BERT(no FT) + BiLSTM + CRF	0.7892
BERT(no FT) + CTM + BiLSTM + CRF	0.7725

Table 3: Results on Validation Subset of the Dataset

Model	Micro F1
RoBERTa + HDP + BiLSTM + CRF	**0.7980**
RoBERTa + BiLSTM + CRF	0.7809
RoBERTa + CTM + BiLSTM + CRF	0.7593
Rank 1 Model	0.8593
Rank 2 Model	0.8581

Table 4: Results on Test Subset of the Dataset

cleaned text with minimum count 5 and threshold 100. The standard English stopwords are removed using NLTK toolkit[5], and bigrams are formed. The words having POS (part-of-speech) tags as NOUN, ADJ (adjective), VERB and ADV (adverb) are lemmatized but proper nouns are retained without any change. POS tagging and lemmatization are done using Spacy[6] framework in Python. All other tokens are discarded. Settings for training HDP model are given in Table 5.

Experiment Setting	Value
Term Weighing Scheme	IDF
minimum collection frequency	5
gamma	1
alpha	0.1
initial k	10
Burn-in Samples	50
Iterations	5000

Table 5: HDP Model Settings

The redundant topics (called dead topics) were purged, and the K topics learned from data were obtained. The topic proportions of a sentence using this model represent the additional features. To batch the corresponding HDP feature vectors for a document, the vectors are padded with K-dimensional zero vectors so that there are uniform number of sentences per document in that batch. The Correlated Topic Model (CTM) was trained on 30 topics, with minimum collection frequency 5.

The model is trained on the train subset of the given data, and the model checkpoint according to the best performance on validation subset is chosen for testing on the test subset of the dataset.

5 Results

Several experiments were conducted and the system for submission was selected based on the performance on the validation subset of the dataset. The proposed system is denoted as, RoBERTa+HDP+BiLSTM+CRF, where the RoBERTa model has been fine-tuned for MLM task on ILDC dataset. A variation of the proposed scheme, denoted as, RoBERTa+CTM+BiLSTM+CRF, has also been tried. This uses parametric topic model CTM instead of nonparametric topic model HDP, to extract additional features. Similar variations have been tried, using BERT (Devlin et al., 2019).

Table 3 gives the results of the different variations of the proposed scheme obtained on the validation subset. The metric used for evaluation in this table is Micro-F1 score. Some new variations of the proposed scheme have been indicated using the suffix '(no FT)' indicating that the RoBERTa

[5]https://www.nltk.org/
[6]https://spacy.io/

model used in these schemes have not been fine-tuned for the MLM task on ILDC dataset.

The results obtained on the test subset of the dataset for the proposed scheme are given in Table 4. The results were obtained by submission of generated predictions on the task platform CodaLab[7]. These results are compared with the scores obtained by the teams securing the first and second positions on the task leaderboard.

The best performance is observed in case of the proposed system. It is also observed that adding HDP features improve the overall performance of the system. Further, the performance is better in case of using nonparametric topic model HDP, than using parametric topic model CTM, for extracting additional features.

6 Conclusion

This work proposes a system for the task of Rhetorical Role prediction in legal documents which can be reformulated as a sequential sentence classification task. The proposed system combines topic modeling and RoBERTa to get sentence representations for each sentence in a document. This is followed by a BiLSTM layer to get contextualised sentence representations. The final layer is a CRF layer which takes into account the dependencies between labels to generate sequential rhetorical role predictions for each document. This paper discusses in the detail the system description of the proposed scheme, along with the experimental setup for reproducibility.

The novelty of the proposed approach lies in the use of topic modeling to augment the sentence representation. This paper discussed results corresponding to different ablations of the proposed approach. One possible future direction of research can be to explore other ways to incorporate the corpus information gained from topic modeling, to improve the task of sequential sentence classification.

References

Paheli Bhattacharya, Shounak Paul, Kripabandhu Ghosh, Saptarshi Ghosh, and Adam Zachary Wyner. 2019. Identification of rhetorical roles of sentences in indian legal judgments. In *International Conference on Legal Knowledge and Information Systems*.

Arman Cohan, Iz Beltagy, Daniel King, Bhavana Dalvi, and Dan Weld. 2019. Pretrained language models for sequential sentence classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3693–3699, Hong Kong, China. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Di Jin and Peter Szolovits. 2018. Hierarchical neural networks for sequential sentence classification in medical scientific abstracts. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3100–3109, Brussels, Belgium. Association for Computational Linguistics.

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*, pages 4420–4429, Marseille, France. European Language Resources Association.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization.

John Lafferty and David Blei. 2005. Correlated topic models. In *Advances in Neural Information Processing Systems*, volume 18. MIT Press.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

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*, abs/1907.11692.

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

Vijit Malik, Rishabh Sanjay, Shubham Kumar Nigam, Kripabandhu Ghosh, Shouvik Kumar Guha, Arnab Bhattacharya, and Ashutosh Modi. 2021b. 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*

[7] https://codalab.lisn.upsaclay.fr/

Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 4046–4062, Online. Association for Computational Linguistics.

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).

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. Curran Associates Inc., Red Hook, NY, USA.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en.

Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. 2006. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581.

A. Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.