VTCC-NLP at SemEval-2023 Task 6: Long-Text Representation Based on Graph Neural Network for Rhetorical Roles Prediction

Huu Hiep Nguyen† and **Hoang Ngo**† and **Khac-Hoai Nam Bui***
Viettel Cyberspace Center, Viettel Group, Vietnam
{hiepnh1307,ngoviethoang735}@gmail.com, nambkh@viettel.com.vn

Abstract

Rhetorical Roles (RR) prediction is to predict the label of each sentence in legal documents, which is regarded as an emergent task for legal document understanding. In this study, we present a novel method for the RR task by exploiting the long context representation. Specifically, legal documents are known as long texts, in which previous works have no ability to consider the inherent dependencies among sentences. In this paper, we propose GNNRR (Graph Neural Network for Rhetorical Roles Prediction), which is able to model the cross-information for long texts. Furthermore, we develop multitask learning by incorporating label shift prediction (LSP) for segmenting a legal document. The proposed model is evaluated on the SemEval 2023 Task 6 - Legal Eval Understanding Legal Texts for RR sub-task. Accordingly, our method achieves the top 4 in the public leaderboard of the sub-task. Our source code is available for further investigation[1].

1 Introduction

SemEval 2023 Task 6 Legal Eval Understanding Legal Texts (Modi et al., 2023) aims to automatically process some intermediate tasks to boost the productivity of legal systems. The task is split into three sub-tasks, which are Rhetorical Roles (RR), Legal Named Entity Recognition (L-NER), and Court Judgement Prediction with Explanation (CJPE). The purpose of the RR prediction task is to standardize an unstructured corpus so that it is easier for the machine to automatically understand legal documents. Participants are given a dataset including long and unstructured legal documents. Each document is segmented into sentences using automatic tools (Kalamkar et al., 2022a). Segmented units could be classified into 13 labels, including 12 semantic labels and a NONE label for those not belonging to any semantic role. The RR task is considered as the sequence classification task with single-label multiple classes, where participants are required to predict the suitable labels for a series of consecutive text segments in those 13 labels. Traditional approach (Saravanan et al., 2008) uses CRF with hand-crafted features to segment the document into seven different roles. (Bhattacharya et al., 2019) proposed BiLSTM-CRF model with sentence embeddings constructed from sent2vec to label rhetorical roles in Indian Supreme Court documents. A multitask learning-based model developed by (Malik et al., 2022) uses label shift information to predict labels. (Kalamkar et al., 2022b) created a RR corpus of English Indian legal documents and proposed the model based on SciBERT-HSLN architecture (Brack et al., 2021). Despite the success of the aforementioned models, there are still two remaining challenges: long-range dependence between sentences and label ambiguity. It is because legal documents are quite long. Therefore, it is hard to represent the contextual information in each sentence. Besides, the performance is negatively affected due to the similar labels such as *ARG_PETITIONER* and *ARG_RESPONDENT*, *PRE_NOT_RELIED* and *PRE_RELIED*.

In order to address the aforementioned problems, we propose a GNN-based Context Representation module to exploit the inter-sentence relations for the long-text representation. For similar labels, we use contrastive learning to learn embedding space. We also use multitask learning framework introduced by (Malik et al., 2022) to boost the model performance. Our final submission for sub-task RR achieved a micro-averaged F1 score of 0.8389 positionings our team in the top 4 on the public leaderboard of the sub-task.

2 Methodology

Our approach is based on the framework of Multitask learning (Malik et al., 2022). Specifically,

† Equal contribution; * Corresponding author

[1]https://github.com/hiepnh137/SemEval2023-Task6-Rhetorical-Roles

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1121–1126
July 13-14, 2023 ©2023 Association for Computational Linguistics

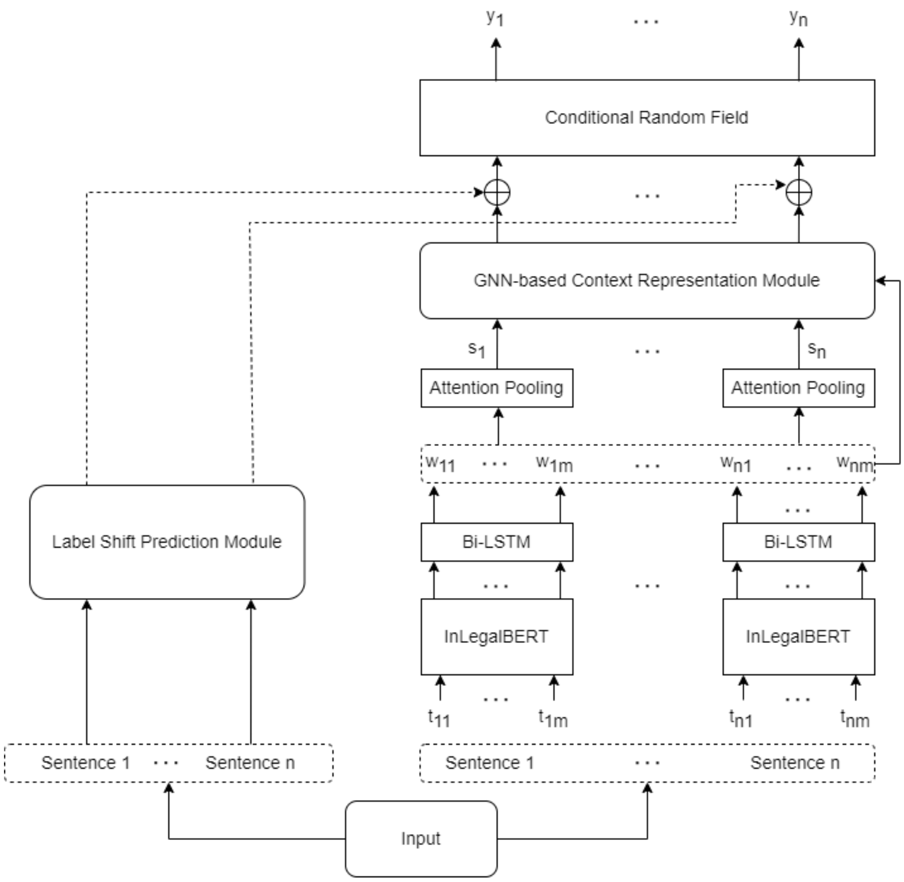

Figure 1: System Architecture of GNNRR model

.

the architecture of our model consists of two main components: the label shift prediction (LSP) component and the rhetorical roles prediction (RRP) component as shown in Figure 1. Specifically, given a legal document $\mathcal{D}$ containing n sentences $\{s_1, s_2, ..., s_n\}$, the label shift prediction task can be seen as an auxiliary task to model relationships between two adjacent sentences and predict whether the labels of them are different ($y_i^{lsp} = 1$) or not ($y_i^{lsp} = 0$). We model LSP via LSP-BiLSTM-CRF architecture, which has shown promising results (Malik et al., 2022). Particularly, supporting $S^{lsp} = (s_1^{lsp}, ..., s_n^{lsp})$ denotes sentence representations, using LSP task can be formulated as follows:

$$E^{lsp} = (e_1^{lsp}, ..., e_n^{lsp}) = BiLSTM(s_1^{lsp}, ..., s_n^{lsp})$$
$$\hat{Y}^{lsp} = CRF(W_o^{lsp} E^{lsp} + b^{lsp}) \tag{1}$$

where E^{lsp} denotes the hidden state and $Y^{lsp} = \hat{y}_1^{lsp}, .., \hat{y}_n^{lsp})$ is the final prediction. W_o^{lsp} is the learning parameter.

On the other hand, the rhetorical roles prediction task can be defined as the sequence labeling task. Our architecture for this task is based on SciBERT-HSLN (Kalamkar et al., 2022b), which includes three main modules: Sentence Encoder, GNN-based Context Representation Module, and Output Layer.

2.1 Sentence Encoder

Given a sentence s_i consisting m sub-words t_{i1},...,t_{im}, we adopt InLegalBERT (Paul et al., 2022) and BiLSTM to capture the contextual information w_{i1},...,w_{im} ($w_{ij} \in \mathbb{R}^d$):

$$w_{i1}, ..., w_{im} = BiLSTM(BERT(t_{i1}, ..., t_{im})) \tag{2}$$

Then, k-head attention pooling (Kalamkar et al., 2022b) is used to produce the corresponding sentence embedding $s_i \in \mathbb{R}^d$:

$$s_i = AttPooling(w_{i1}, ..., w_{im}) \tag{3}$$

2.2 GNN-based Context Representation

The main challenge of the RR task is that the document is very long length (around 3000 words in each document). In this regard, with only 247 documents for training, representing the document with full attention seem not effective. Therefore, in this study, we propose a spare attention method for representing the long text using a graph neural network (GNN), which has proved promising results for long text representation (Phan et al., 2022). Accordingly, the graph structure to model the relationship between words in a document can be constructed based on two following steps:

- Directed edges for connecting sequential words in a sentence.
- Undirected edges between the words with the same entities, which are extracted by the baseline model in the sub-task Legal-NER[2].

Figure 2 illustrates the architecture of the proposed module for the long-text representations. The initialized node embedding H_w^0 is calculated by averaging its sub-word embeddings w_i (Eq. 2), which is formulated as follows:

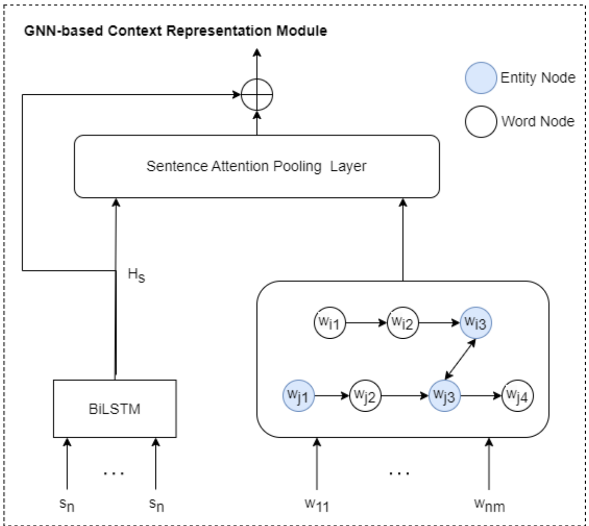

Figure 2: GNN-based Context Representation Module

.

$$H_w^0 = MeanPooling(w_{11}, ..., w_{nm}) \quad (4)$$

Subsequently, the node embeddings are iteratively updated via multi-head Gaph Attention Network (GAT) (Velickovic et al., 2018) and Feed Forward layer (FFN):

$$\begin{aligned} U_w^t &= GAT(H_w^{t-1}, H_w^{t-1}, H_w^{t-1}) \\ H_w^t &= FFN(U_w^{t-1} + H_w^{t-1}) \end{aligned} \quad (5)$$

where t represents the t^{th} iteration.

[2]https://github.com/Legal-NLP-EkStep/legal_NER

In this regard, sentence embedding is enriched by different levels of information including words and sentences, which can be processed in two steps: i) using BiLSTM for generating the hidden state of the sentence; ii) the hidden state of sentences and words (Eq.5) are then used for updating the new sentence representation. Specifically, the process can be sequentially formulated as follow:

$$\begin{aligned} h_{s_1}, ..., h_{s_n} &= BiLSTM(s_1, ..., s_n) \\ h'_{s_i} &= Attention(h_{s_i}, \{h_{w_{i1}}, ..., h_{w_{in}}\}) + h_{s_i} \\ e_i^{rr} &= h_{s_i} \oplus h'_{s_i} \end{aligned} \quad (6)$$

where $\oplus$ stands for concatenate function, e_i^{rr} denotes the new representation of sentence i^{th}.

2.3 Output Layer

The output of new sentence representations E^{rr} are then concatenated with the hidden state of LSP task E^{lsp} before put into the output layer:

$$e_i = e_i^{lsp} \oplus e_i^{rr} \quad (7)$$

where e_i denotes the final representation of sentence i^{th}. In particular, for the output layer, we transform the hidden representations of sentences to the logits via a linear transformation. Furthermore, CRF is utilized to improve performance. The process can be formulated as:

$$\hat{Y}^{rr} = CRF(W_o E + b) \quad (8)$$

where $E = \{e_1, .., e_n\}$ denotes the sentence embedding. W_o and b are trainable parameters. $\hat{Y}^{rr} = (\hat{y}_1^{rr}, ..., \hat{y}_n^{rr})$ is the rhetorical role prediction.

2.4 Training Process

Following (Malik et al., 2022), we jointly optimize LSP and RR tasks. The objective of the LSP task is formulated as follows:

$$\mathcal{L}^{lsp} = -\sum_{i=1}^{n} \Big(y_i^{lsp} log(\hat{y}_i^{lsp}) + (1 - y_i^{lsp}) log(1 - \hat{y}_i^{lsp}) \Big) \quad (9)$$

Similarly, the objective of the RR prediction task is calculated as follows:

$$\mathcal{L}^{rr} = -\sum_{i=1}^{n} \sum_{l=1}^{L} \left(y_{i,l}^{rr} log(\hat{y}_{i,l}^{rr}) \right) \quad (10)$$

where L denotes the number of rhetorical role labels, $Y_i^{rr} = (y_{i,1}^{rr}, ..., y_{i,L}^{rr})$ represents the gold label

Model	Dev		Test
	Micro-F1	Weighted-F1	Micro-F1
w/o contrastive & multitask	83.61	82.82	-
w/o contrastive & graph	83.43	82.46	-
w/o contrastive	83.74	82.91	82.71
GNNRR	**84.30**	**83.60**	**83.89**

Table 1: The main results of the proposed model

of RR prediction task. The final objective is then formulated as:

$$\mathcal{L} = \lambda_1 \mathcal{L}^{lsp} + \lambda_2 \mathcal{L}^{rr} \tag{11}$$

where λ_1 and λ_2 are hyperparameters.

Contrastive Learning

Long legal documents contain various sentences which have ambiguous labels. Some similar pairs of labels such as (*ARG_PETITIONER* and *ARG_RESPONDENT*), (*PRE_NOT_RELIED* and *PRE_RELIED*) hinder the autonomous understanding process. For a better solution to the ambiguity problem, we utilize the Supervised Contrastive Learning (SupCon) method (Khosla et al., 2020) to enhance hidden representation among labels.

Algorithm 1: Supervised Contrastive Learning example selection algorithm

Data: List of sentence embeddings $\{h\}_i$,
List of clusters C,
Contrastive batch size B,
Number of examples per label k
Result: Supervised Constrastive loss $\mathcal{L}^{sc}$
$\mathcal{I} \leftarrow \varnothing$;
for j *in* $[1..B]$ **do**
    c = Random(C) ; `/*Randomly select a cluster*/`
    $s_1, s_2, ..., s_k$ = Sample(c, k) ; `/*Randomly select k examples*/`
    Add($\mathcal{I}$, $\{s\}_{i=1}^{k}$)
end
$\mathcal{L} = SupConLoss(\mathcal{S})$

Algorithm 1 explains the sampling process of SupCon. Sentences in a document are sorted into 13 clusters based on their labels. Repeat these steps to form a training batch: (i) Randomly select a cluster. (ii) Pick k sentences from the cluster, then add them to the batch. For clusters with less than k example, we re-sample and use a Dropout mask to make them alternative views of the original examples. Given a formed batch, we generate a positive and negative mask for each anchor example. Eventually, SupCon loss is defined as follows:

$$\mathcal{L}^{sc} = \sum_{i \in \mathcal{I}} \frac{-1}{|P(i)|} \sum_{p \in P(i)} log \frac{exp(z_i \cdot z_p / \tau)}{\sum_{a \in N(i)} exp(z_i \cdot z_a / \tau)} \tag{12}$$

where $\mathcal{I}$ is the batch, $z_i = We_i$ is hidden representation of i-th sentence, $N(i)$ is set of negative samples of i-th sample, $P(i)$ is set of positive samples of i-th sample, | . | stands for cardinality and τ is temperature scaling parameter.

In this regard, the loss function (Eq.11) can be re-calculated as follows:

$$\mathcal{L} = \lambda_1 \mathcal{L}^{lsp} + \lambda_2 \mathcal{L}^{rr} + \lambda_3 \mathcal{L}^{sc} \tag{13}$$

where λ_1, λ_2 and λ_3 are hyperparameters.

3 Experiment

3.1 Hyperparameter Setting

For the GNN setting, the number of GAT layers is set to 3 and each layer includes 4 heads. λ_1, λ_2 and λ_3 are set to 0.55, 0.35 and 0.1, respectively. We use a batch size of 32 and apply Adam (Kingma and Ba, 2015) with a learning rate 3e-5 to optimize the parameters in our model. For contrastive learning, the batch size B is set to 16, and the number of sentences k is set to 2. All experiments are conducted on a single NVIDIA A100 card.

3.2 Main results

Table 1 reports the main results of our model on the SemEval 2023 Task 6 - Legal Eval Understanding Legal Texts for RR sub-task. Specifically, we execute the experiment with four simplified versions of the proposed model such as i) w/o contrastive learning and multitask learning; ii) w/o contrastive learning and graph modules; iii) w/o contrastive

Model	Weighted-F1
SciBert-HSLN	79.47
SciBert-HSLN + InLegalBERT	-
Multitask + BERT	80.26
Multitask + InLegalBERT	80.86
HeterSum + BERT	80.02
HeterSum + InLegalBERT	
Entity-based Graph + BERT	80.39
Entity-based Graph + InLegalBERT	80.58
Multitask + Entity-based Graph + BERT	80.78
Multitask+ Entity-based Graph + InLegalBERT	81.80
Multitask + Entity-based Graph + Contrastive + BERT	-
Multitask + Entity-based Graph + Contrastive + InLegalBERT (GNNRR)	**82.79**

Table 2: RR prediction task with different approaches on the dev dataset. Note that in this experiment, all the models do not finetune the pre-trained model.

learning; and iv) our full GNNRR model. As result, our full GNNRR model achieves the F1 score of 84.3 % and 83.89 % on the dev and test set, respectively, which belongs to the top 4 on the public leaderboard (top 1 archives 85.93 % of the test set). Furthermore, the results of the full GNNRR model outperform variant versions indicating the importance of each module.

3.3 Results Analysis and Discussion

We further investigate the effectiveness of the proposed model by trying to exploit different emergent approaches for this task, which are sequentially described as follows:

- SciBERT-HSLN (Kalamkar et al., 2022b): The baseline model, which is presented in the respective paper of the benchmark dataset. Specifically, the model is developed based on a hierarchical sequential labeling network.
- Multitask Learning (Malik et al., 2022): The method that uses the multi-task learning framework, where LSP is an auxiliary task. In the main task, we use the SciBERT-HSLN model for prediction.
- HeterGNN (Wang et al., 2020): The model that replaces the GNN-based Context Representation Module with a Heterogeneous Graph Neural Network of words and sentences. In this graph, common words are intermediaries for connecting sentences.
- Entity-based Graph: The model is based on SciBERT-HSLN where the GNN-based Context Representation Module is utilized for enriching contextual information of sentences instead of the context enrichment module in the original model.
- Multitask + Entity-based Graph: The model that uses both Multi-task Learning and Entity-based Graph
- Multitask + Entity-based Graph + Contrastive: The full our proposed GNNRR model

Table 2 shows the results of our model compared with different approaches. Accordingly, we make the following observations: i) using the pre-trained language model InLegalBERT achieves better results over BERT ii) using entities for constructing graph structure is able to improve the performance (80.39 % vs 80.02 %); iii) designing multitask learning by incorporating label shift as an auxiliary task is able to boost the performance (80.39 % vs 79.47 %); iv) our model, which includes multitask learning and entity-based GNN for enriching contextual information is able to achieve the best performance.

4 Conclusion

In this paper, we represent our methodology for the Rhetorical Roles Predictions task in SemEval 2023 Task 6 Legal Eval Understanding Legal Texts. Our approach, which is a combination of using Multitask Learning, Contrastive Learning, and the proposed GNN-based Context Representation Module, is proven effective via experiments. Moreover, our best result ranks top fourth in the public leaderboard of that sub-task.

References

Paheli Bhattacharya, Shounak Paul, Kripabandhu Ghosh, Saptarshi Ghosh, and Adam Wyner. 2019. Identification of rhetorical roles of sentences in indian legal judgments. *CoRR*, abs/1911.05405.

Arthur Brack, Anett Hoppe, Pascal Buschermöhle, and Ralph Ewerth. 2021. Sequential sentence classification in research papers using cross-domain multi-task learning. *CoRR*, abs/2102.06008.

Prathamesh Kalamkar, Aman Tiwari, Astha Agarwal, Saurabh Karn, Smita Gupta, Vivek Raghavan, and Ashutosh Modi. 2022a. 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.

Prathamesh Kalamkar, Aman Tiwari, Astha Agarwal, Saurabh Karn, Smita Gupta, Vivek Raghavan, and Ashutosh Modi. 2022b. Corpus for automatic structuring of legal documents. *CoRR*, abs/2201.13125.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *CoRR*, abs/2004.11362.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Vijit Malik, Rishabh Sanjay, Shouvik Kumar Guha, Angshuman Hazarika, Shubham Nigam, Arnab Bhattacharya, and Ashutosh Modi. 2022. Semantic segmentation of legal documents via rhetorical roles. In *Proceedings of the Natural Legal Language Processing Workshop 2022*, pages 153–171, Abu Dhabi, United Arab Emirates (Hybrid). 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).

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

Tuan-Anh Phan, Ngoc-Dung Ngoc Nguyen, and Khac-Hoai Nam Bui. 2022. Hetergraphlongsum: Heterogeneous graph neural network with passage aggregation for extractive long document summarization. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 6248–6258. International Committee on Computational Linguistics.

M. Saravanan, B. 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*.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Danqing Wang, Pengfei Liu, Yining Zheng, Xipeng Qiu, and Xuanjing Huang. 2020. Heterogeneous graph neural networks for extractive document summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6209–6219, Online. Association for Computational Linguistics.