AntContentTech at SemEval-2023 Task 6: Domain-adaptive Pretraining and Auxiliary-task Learning for Understanding Indian Legal Texts

Jingjing Huo, Kezun Zhang, Zhengyong Liu, Xuan Lin*, Wenqiang Xu
Maozong Zheng, Zhaoguo Wang, Song Li
FinContentTech, Ant Group
{huojingjing.hjj, kezun.zkz, liuzhengyong.lzy, daxuan.lx, yugong.xwq,
zhengmaozong.zmz, wangzhaoguo.wzg, hejian.ls}@antgroup.com

Abstract

The objective of this shared task is to gain an understanding of legal texts, and it is beset with difficulties such as the comprehension of lengthy noisy legal documents, domain specificity as well as the scarcity of annotated data. To address these challenges, we propose a system that employs a hierarchical model and integrates domain-adaptive pretraining, data augmentation, and auxiliary-task learning techniques. Moreover, to enhance generalization and robustness, we ensemble the models that utilize these diverse techniques. Our system ranked first on the RR sub-task and in the middle for the other two sub-tasks. Our code is publicly available here[1].

1 Introduction

Certain heavily populated nations, such as India, are faced with a surfeit of unresolved legal cases, and the development of automated AI tools can potentially aid legal practitioners in expediting the judicial system. Due to the intricate nature of the judicial process, the SemEval-2023 Task 6 (Modi et al., 2023) proposes three sub-tasks that can serve as foundational components for creating legal AI applications, *i.e.* Sub-task A: Rhetorical Roles Prediction (RR), Sub-task B: Legal Named Entities Extraction (L-NER) and Sub-task C: Court Judgement Prediction with Explanation (CJPE).

The RR sub-task is to organize unstructured legal documents into semantically cohesive units (classification at the sentence level), while the second L-NER sub-task involves identifying relevant entities within a legal document. The third CJPE sub-task focuses on predicting the outcome of a judgment (classification at the document level) and providing a corresponding explanation.

The task at hand focusing on Indian legal documents is inherently challenging. Firstly, legal documents exhibit considerably greater length, and a higher degree of unstructured content and noise when compared to shorter documents used for the training of traditional natural language processing models. Moreover, the presence of legal-specific lexicons within the documents makes general pretrained neural models ineffective. Further, legal provisions vary from one nation to another, and the legal domain also has several sub-domains that align with different legal provisions. It follows that a pretrained model equipped with legal knowledge specific to one sub-domain may not be applicable to another distinct sub-domain. Another key challenge in this task is the limited size of annotated data due to the time-consuming and expensive annotation process (Kalamkar et al., 2022).

To address the aforementioned challenges, we propose a system based on the baseline model called SciBERT-HSLN in (Kalamkar et al., 2022; Brack et al., 2021) and equipped with domain-adaptive pretraining, data augmentation strategies, as well as auxiliary-task learning techniques. In summary, we list here our findings that are derived from the experimentation:

- Continuing to pretrain an already powerful legal-domain LM with Indian legal text and task-specific data can further boost the performance of our system.
- Data augmentation strategies can substantially enhance the performance of the system due to the limited size of annotated data. However, it is important to note that exceeding a certain threshold for augmentation data may not lead to proportional improvements in performance.
- Integrating auxiliary tasks such as BIOE tagging, supervised contrastive learning, or prototypical contrastive learning can indeed enhance the system's performance, but this

*Corresponding author
[1] https://github.com/ContentTech/rhetorical-role-baseline

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 402–408

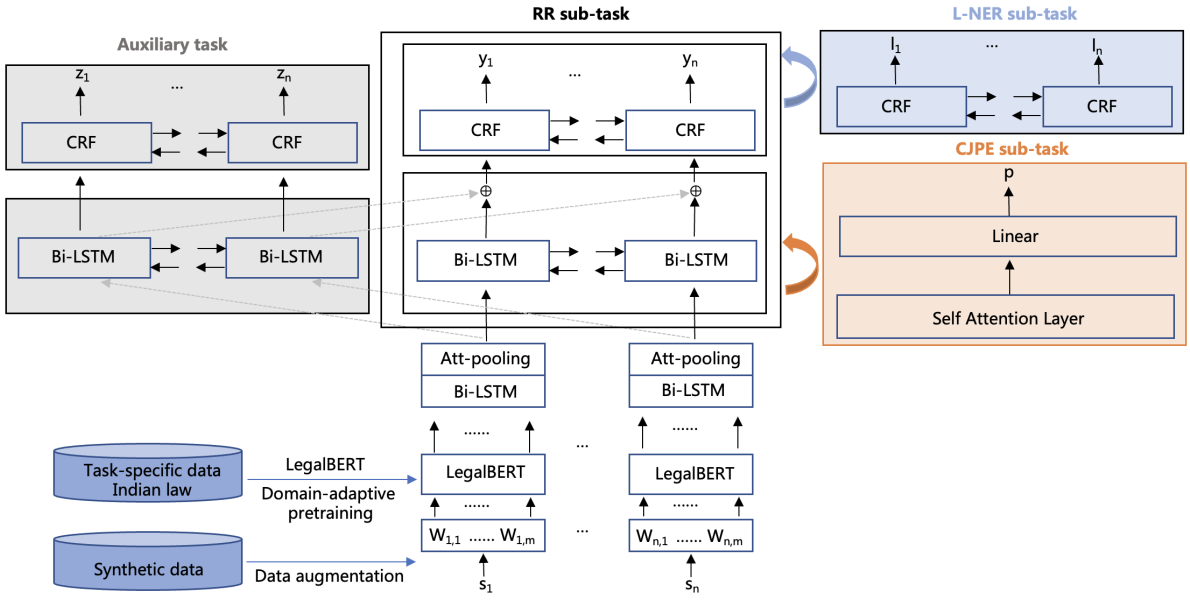

Figure 1: The overall architecture of our proposed system, inspired by (Kalamkar et al., 2022; Brack et al., 2021).

may not always be consistent across different datasets.

2 System overview

Given a legal case, D, containing the sentences $[s_1, s_2, ...s_n]$, the RR sub-task is to predict the label y_i for each sentence $s_i \in D$. We consider it as a sequence labeling task and take the SciBERT-HSLN architecture (Brack et al., 2021) as the backbone of our system, where each sentence is fed into a pretrained model to get word embeddings, these embeddings are further passed through a word Bi-LSTM layer (Hochreiter and Schmidhuber, 1997) followed by an attention-based pooling layer to get sentence representations $s'_1, s'_2, ...s'_n$.

For RR sub-task, these sentence representations are subsequently processed by another Bi-LSTM, resulting in contextualized sentence representations, $c_1, c_2, ...c_n$, with a final CRF (Lafferty et al., 2001) layer utilized to give the prediction of RR. The non-colored part of Figure 1 illustrates the backbone of this architecture. We enhance our system with domain-adaptive pretraining, data augmentation, and auxiliary-task learning techniques. Further details of these techniques are discussed in the subsequent subsections.

In the case of L-NER sub-task, sentence representations are directly fed into a CRF layer to do the span classification. And for CJPE sub-task, we prepend a global token to the input sequence and use a self-attention layer instead of the Bi-LSTM to generate contextualized sentence representations. The global token attends to all other sentences and is utilized to generate the final prediction of the given judgement case, while the attention scores are used to extract explanations.

2.1 Domain-adaptive Pretraining

Some previous works have shown the ability of pretrained LMs to facilitate transfer learning and highlighted the benefit of continued pretraining with domain-specific unlabeled data (Gururangan et al., 2020). In our task, we argue that continuing to pretrain an already powerful legal-domain LM with Indian legal text and task-specific data can result in further improvements in performance.

We, firstly, compare the performance of two LegalBert models, $\text{LegalBert}_{\text{zlucia}}$ (Zheng et al., 2021) and $\text{LegalBert}_{\text{nlpaueb}}$ (Chalkidis et al., 2020), and found that $\text{LegalBert}_{\text{zlucia}}$, which is initialized with the base BERT model and trained for an additional 1M steps on the masked language modeling and next sentence prediction objective with a pretraining corpus sourced from the entire Harvard Law case corpus from 1965 until the present (37GB), outperformed $\text{LegalBert}_{\text{nlpaueb}}$, which was trained from scratch on a legal corpus with a size of 12GB. Based on this result, $\text{LegalBert}_{\text{zlucia}}$ is chosen for further experimentation.

Furthermore, taking the variance between U.S. law and Indian law into consideration, we investigate the impact of continuing to pretrain Legal-

BERT using Indian legal texts and task-specific data. To do so, we scrape Indian legal cases from the Indian Kanoon website[2] and divide the documents into sentences using SpaCy[3]. This data along with the RR data is then used to further pretrain the $\text{LegalBert}_{\text{zlucia}}$ model. We compared the effectiveness of the pretraining corpora using 20w sentences and 40w sentences.

2.2 Data Augmentation

As can be seen from Table 1, the training set for the RR sub-task is inadequate in size, consisting of only 245 documents and a limited number of samples for certain labels. To address this issue, we employ two kinds of augmenting strategies.

Firstly, at the document level, we aim to enhance consistency within each block and diversify differences between blocks. A block, in this case, refers to consecutive sentences with the same label. To achieve this, we randomly remove 10% of the blocks and replace 40% of them with blocks of the same label from other documents. We also shuffle the sentences within the blocks. At the sentence level, we propose that replacing a semantic unit is more effective than merely replacing a sentence with another sentence of the same label. Therefore, we employ the TreeMix system (Pickrell and Pritchard, 2012) on all sentences in the corpus, 40% of which the sub-trees are randomly replaced with those of the same type extracted from sentences with the same label.

Secondly, we utilize back-translation to further augment the data. We select German as the intermediate language due to its linguistic similarities with English. Both are Germanic languages and share a significant amount of vocabulary, grammatical structure, and syntax. For bi-directional translation models, we choose wmt19-en-de and wmt19-de-en (Ng et al., 2020), both of which have been reported to demonstrate state-of-the-art performance on the WMT'19 shared task. These two strategies double the original dataset size and are denoted as BLOCK and BACK, respectively.

Additionally, since the use of any publicly available data is allowed by the task organizer, we also find two public corpora proposed by the previous similar tasks (Ghosh and Wyner, 2019; Malik et al., 2021) that possess partially overlapping label sets with the target task and merged them with the datasets used in the RR sub-task. Despite the fact that their label definitions do not precisely align with those of the RR task, they are deemed to be beneficial as supplementary data for the task at hand after a careful screening process. This corpus is referred to as **O2T** in the table.

Labels	RR	O2T	BLOCK	BACK
ANALYSIS	10695	10695	20206	21294
ARG_PET	1315	1315	2455	2615
ARG_RES	698	698	1309	1390
FAC	5744	12949	10769	11432
ISSUE	367	504	649	725
NONE	1423	1786	2564	2768
PREAMBLE	4167	4167	7517	8028
PRE_NOT	158	827	291	315
PRE_RELIED	1431	3884	2721	2856
RATIO	674	9405	1266	1346
RLC	752	2146	1416	1487
RPC	1081	1748	1975	2140
STA	481	2466	887	948
NumSent	28980	52590	54025	57344
NumCases	245	395	490	490

Table 1: Train dataset statistics of RR sub-task.

2.3 Auxiliary Tasks

There is a high likelihood that consecutive sentences in legal cases share the same rhetorical role labels. To facilitate RR classification, we introduce the **BIOE** tagging, wherein the "B", "I", and "E" labels are assigned to the first, intermediate, and last sentence in a block, respectively, while sentences with the original label "None" are assigned the "O" label. This is inspired by that we observed the baseline model occasionally makes mistakes in the middle of a block.

Moreover, we notice that the baseline model frequently confuses certain labels (*e.g.* FAC/ANALYSIS/ISSUE, ARG_PET/ARG_RES *etc.*). To address this issue, we integrate the supervised contrastive loss, which aims to train a representation space that can better distinguish between labels by contrasting positive and negative examples, into our system. The definition of supervised contrastive loss (**SCL**) (Khosla et al., 2020) is provided below:

$$L^{\text{sup}} = \sum_{i=1}^{2N} \frac{-1}{2N_{\tilde{y}_i} - 1} \sum_{i=1}^{2N} \mathbb{I}_{i \neq j} \mathbb{I}_{\tilde{y}_i = \tilde{y}_j} \text{sim}(c_i, c_j)$$

$$\text{sim}(c_i, c_j) = \log \frac{\exp(c_i \cdot c_p / \tau)}{\sum_{k=1}^{2N} \mathbb{I}_{i \neq k} \exp(c_i \cdot c_k / \tau)}$$

Here, N is the number of sentences in a batch, c_i is

[2] https://indiankanoon.org/
[3] https://spacy.io/

Data	RR	BIOE	SCL	ALL
D0	81.33	81.40	**81.89**	80.85
D1	80.67	79.90	80.84	**81.59**
D2	75.61	**76.72**	76.26	76.17
D3	81.82	82.11	80.92	**82.60**
D4	78.09	78.52	78.40	**78.78**

Table 2: The performance of different combinations of auxiliary tasks on splitting datasets, the best-performing models on each dataset are selected for ensembling in RR sub-task.

Pretrained Model	Dev F1
Bert	82.06
Deberta	82.13
SciBert	78.02
$\text{LegalBert}_{\text{nlpaueb}}$	82.81
$\text{LegalBert}_{\text{zlucia}}$	83.01
$\text{LegalBert}_{\text{20w}}$	83.65
$\text{LegalBert}_{\text{40w}}$	83.12

Table 3: The performance of pretrained Models in RR sub-task.

contextualized sentence embedding of i-th sample, and $(N+i)$-th sample in the batch is the augmented sample originating from the i-th sample by simply applying dropout. $N_{\tilde{y}_i}$ is the total number of sentences in the batch that have the same label $\tilde{y}_i$. τ is the temperature normalization factor.

Furthermore, we also utilize prototypical contrastive learning (PCL) (Li et al., 2021) to enhance the representation of our system by emphasizing the similarity between a sample and its label compared to other labels and assigning discounted weights for labels according to their similarities to denoise label confusion. Specifically, we represent each label_j using a prototype vector p_j and calculate its similarity with another label prototype vector $\text{p}_{j'}$ using the cosine function. We ranked the similarities in descending order as $\text{rank}_{j'}$, indicating that the label at $\text{rank}_{j'}$ is label_j's j'-th most similar label. As a result, the similarity between c_i and label vector p_j is the cosine score represented as $f(c_i, p_j)$. The discounted similarity and the prototypical contrastive loss are denoted as $f^{'}(c_i, p_j)$ and L^{pcl}, separately:

$$f^{'}(c_i, p_j) = \frac{f(c_i, p_j)}{\log_2(\text{rank}_j + 1)}$$

$$L^{\text{pcl}}_{x_i} = -\log \frac{e^{f^{'}(c_i, p^+)/\tau}}{\sum_{j=1}^{J} e^{f^{'}(c_i, p_j)/\tau}}$$

where c_i is the representation of sample x_i, p^+ is the ground truth label vector, J is the number of labels, and L^{pcl} summarizes $L^{\text{pcl}}_{x_i}$ of all samples.

2.4 Ensemble Model

As stated in the introduction, we utilize model ensembling to enhance the robustness of our system. To achieve this, we combine the training set and development set and divide it into five separate folds, with four of these folds utilized as the training set, and the remaining fold as the new development set, by which we get five datasets. We train our system on each of these datasets using different combinations of auxiliary tasks, and then simply take the majority voting of the top-performing models on each dataset for the final prediction.

3 Training Setup

For fine-tuning our system, we nearly follow the hyperparameter configurations utilized in SciBERT-HSLN (Brack et al., 2021), with the exception of setting the learning rate to 5e-04, dropout rate to 0.1, Bi-LSTM hidden size to 768, and accumulation steps to 2. We use these hyperparameters in all of our experiments, with the maximum number of epochs and early stopping epochs set to 70 and 10, respectively. The training process is executed on NVidia V100 GPUs, and all the performance is evaluated on the development set released by the organizer, except for the final resembling.

4 Main Results

Finally, our system got an F1 of score 85.93 on the Leaderboard and ranked 1st in RR sub-task. In Table 2, we highlight the five models that we ensemble for the final winning system. We observe that integrating auxiliary tasks (BIOE, SCL, or both) indeed has positive effects, but various performances across different datasets, and consequently we choose the best model for each dataset for ensembling.

4.1 How significant is the domain-adaptive pretraining?

RR sub-task Table 3 compares the performance of using different pretrained models, *i.e.* Bert, Deberta, and SciBert, as well as various versions of LegalBert. $\text{LegalBert}_{\text{20w}}$ and $\text{LegalBert}_{\text{40w}}$ are

RR	O2T	BLOCK	BACK	Dev F1
✓				83.65
✓	✓			84.47
✓		✓		84.31
✓			×1	84.30
✓			×2	84.13
✓	✓		×1	84.99
✓	✓	✓	×1	85.06

Table 4: The performance of different data augmentation strategies in RR sub-task.

higher versions of $\text{LegalBert}_{\text{zlucia}}$ that are continued to be trained with 20w and 40w sentences extracted from Indian legal cases, respectively. As indicated in the table, incorporating Indian legal text can further enhance the performance of a powerful legal-domain Bert. However, we do not observe any additional improvements after doubling the size of the Indian legal text that was used as pertaining data.

CJPE sub-task We also verified the effectiveness of continuing training in the C-1 judgment prediction task. Our experiments show that using continued training $\text{LegalBert}_{\text{20w}}$ leads to a significant improvement in performance on the test set released by the organizer, with an F1 score of 70.58 compared to 68.16 achieved using $\text{LegalBert}_{\text{zlucia}}$. And selecting the top 30% of sentences based on their attention scores provided by the first token give us a ROUGE-2 of 0.0445 on Task C-2 on the Leaderboard.

4.2 How significant is augmented data?

RR sub-task In this Section, we analyze the impact of different data augmentation strategies on the performance of our system, as presented in Table 4. Incorporating O2T, BLOCK, and BACK separately yields similar significant improvements in performance. However, we observe that doubling the size of BACK does not provide any additional improvements and even leads to a slight degradation in performance. The most substantial improvement is observed when combining O2T and BACK, resulting in an F1 score of 84.99. Additionally, further adding BLOCK to this combination resulted in a marginal boost, with an F1 score of 85.06.

4.3 How useful auxiliary-tasks are?

RR sub-task Our experimental results shown in Table 5 indicate that incorporating auxiliary tasks can be beneficial in enhancing the performance of our system in the RR task. Specifically, we observe a slight improvement in performance when incorporating the BIOE tagging or adding prototypical contrastive loss, while a more substantial improvement is achieved when incorporating the supervised contrastive learning task. However, we observe a slightly lower F1 score when combining **BIOE** and **SCL**. The reason for this may be due to the conflicting signals introduced by the two tasks, which can lead to a decrease in overall performance.

RR	BIOE	PCL	SCL	Dev F1
✓				85.06
✓	✓			85.14
✓		✓		85.17
✓			✓	85.57
✓	✓		✓	85.49

Table 5: The performance of different auxiliary tasks in RR sub-task.

Model	Test F1
Bert-CRF	85.90
$\text{Bert-CRF}_{\text{pcl}}$	86.09
$\text{Bert-CRF}_{\text{ensemble}}$	86.22

Table 6: Main results of L-NER sub-task.

L-NER sub-task The main results of L-NER are illustrated in Table 6. The use of the continued pretrained $\text{LegalBert}_{\text{20w}}$ in Bert-CRF serves as the baseline, yielding a test F1 score of 85.9 on the Leaderboard. The integration of the prototypical contrastive loss results in a slight performance improvement, leading to a test F1 score of 86.09. Finally, the submitted result with a test F1 score of 86.22 is obtained through an ensemble of models based on separate training data and multiple auxiliary tasks.

5 Conclusion

In conclusion, the paper proposes a system for the SemEval-2023 task 6, that addresses the challenges of domain-specificity legal documents, and limited annotated data. The system employs a hierarchical model, enhanced by domain-adaptive pretraining, data augmentation, and auxiliary-task learning techniques. The models that utilize these techniques are ensembled to further improve the system's performance. The system ranked first in RR sub-task and achieved a moderate performance for the other two sub-tasks.

Acknowledgements

This work was supported by Ant Group.

References

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

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Jinlan Fu, Xuanjing Huang, and Pengfei Liu. 2021. Spanner: Named entity re-/recognition as span prediction. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*.

Saptarshi Ghosh and Adam Wyner. 2019. Identification of rhetorical roles of sentences in indian legal judgments. In *Legal Knowledge and Information Systems: JURIX 2019: The Thirty-second Annual Conference*, volume 322, page 3. IOS Press.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

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.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673. Curran Associates, Inc.

John D. Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning*.

Junnan Li, Pan Zhou, Caiming Xiong, and Steven C.H. Hoi. 2021. Prototypical contrastive learning of unsupervised representations. *The International Conference on Learning Representations*.

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

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2020. Facebook fair's wmt19 news translation task submission. In *Proc. of WMT*.

Joseph Pickrell and Jonathan Pritchard. 2012. Inference of population splits and mixtures from genome-wide allele frequency data. *Nature Precedings*, pages 1–1.

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. In *Proceedings of the 18th International Conference on Artificial Intelligence and Law*. Association for Computing Machinery.

A Appendix

A.1 Details of ensemble procedure

The ensemble procedures for RR and L-NER subtasks are outlined in Figure 2. For L-NER subtask we utilized four models, including two models trained on both judgment and preamble data, referred to as bert-crf and bert-span(Fu et al., 2021) which label single token or a string of consecutive tokens separately, and the other two models only used judgment or preamble data, which were called judgment-crf and preamble-crf, respectively. Similar to the RR sub-task, we used 5-fold cross-training by dividing the training data into five folds and trained the aforementioned four models on each of them. The ensemble procedure involves two primary steps. Firstly, we average the logits obtained by the same type of model trained on different datasets and use them to predict entities. Secondly, we vote for judgment results by ensembling bert-crf, bert-span, and judgment-crf models, and vote for preamble results by ensembling bert-crf, bert-span, and preamble-crf models.

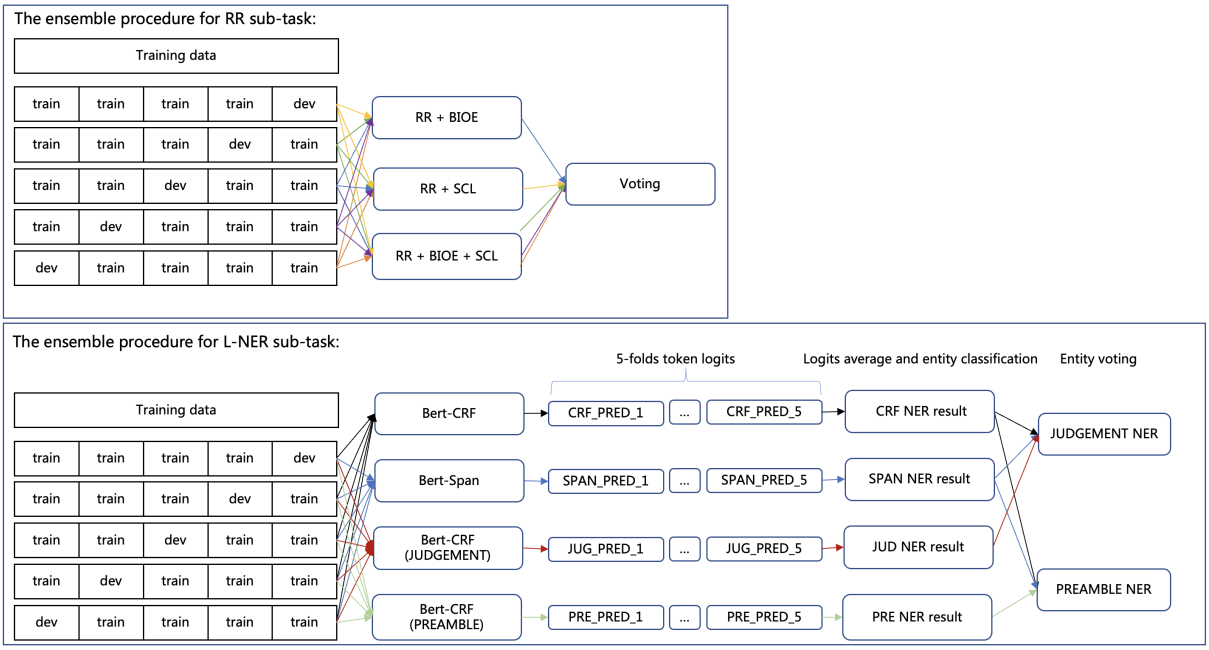

Figure 2: Pipeline of our ensemble method.