nclu_team at SemEval-2023 Task 6: Attention-based Approaches for Large Court Judgement Prediction with Explanation

Nicolay Rusnachenko
Newcastle University
Newcastle Upon Tyne
England
nicolay.rusnachenko@ncl.ac.uk

Thanet Markchom
University of Reading
Reading, England
t.markchom@pgr.reading.ac.uk

Huizhi Liang
Newcastle University
Newcastle Upon Tyne
England
huizhi.liang@ncl.ac.uk

Abstract

Legal documents tend to be large in size. In this paper, we provide an experiment with attention-based approaches complemented by certain document processing techniques for judgment prediction. For the prediction of explanation, we consider this as an extractive text summarization problem based on an output of (1) CNN with attention mechanism and (2) self-attention of language models. Our extensive experiments show that treating document endings at first results in a 2.1% improvement in judgment prediction across all the models. Additional content peeling from non-informative sentences allows an improvement of explanation prediction performance by 4% in the case of attention-based CNN models. The best submissions achieved 8th and 3rd ranks on judgment prediction (C1) and prediction with explanation (C2) tasks respectively among 11 participating teams. The results of our experiments are published[1].

Judicial process involves a lot of challenges to perform a quick and correct case prediction. The significant growth of cases, especially in highly populated countries, significantly leverages the capacity of competent judges' work. The necessity of an automatic assistance system is crucial and becomes a main reason why SemEval-2023 LegalEval (Modi et al., 2023) competition promotes the studies in this area (Kalamkar et al., 2022; Malik et al., 2021). *Court judgment prediction* (CJP) problem is separated into two parts: judgment prediction ("Accepted" or "Denied") and prediction with explanation (CJPE) (Malik et al., 2021).

Within the last few years, deep learning techniques have had significant breakthroughs. Among many significant achievements, it is worth highlighting the appearance of the *attention mechanism* which plays a crucial part in the text generative models commonly found across the whole natural language processing (NLP) domain. Initial studies proposed this mechanism to address a long input sequence of neural machine translation problem (Bahdanau et al., 2014), with further applications in other NLP domains, including text classification (Shen and Huang, 2016; Zhou et al., 2016). The appearance of self-attention (Vaswani et al., 2017) proposes state-of-the-art results for a large set of NLP tasks. Self-attention becomes a backbone component of further models (Devlin et al., 2019), with the appearance of target-oriented transformers. Once weights are visualized, the attention mechanism serves as information for further analysis of what was considered for making a decision. Consequently, our contribution in this paper is conducting experiments with document processing techniques in a combination with attention-based mechanisms for judgment prediction explanation to complement the findings of the task paper (Malik et al., 2021).

This paper is organized as follows. In Section 1, we describe attention mechanisms and approaches with explanation generation algorithms based on them. Section 2 is devoted to resources adopted for conducting experiments. In Section 3, we describe a variety of different document reduction techniques that were adopted in model training. Sections 4 and 5 provide models and dataset processing details with further obtained results.

1 System Description

1.1 CNN with Attention

We adopt convolutional neural network (Zeng et al., 2015) (CNN) for the prediction task C1 and CNN version with attention mechanism focus on words that have a decisive effect on classification (Zhou et al., 2016). In this paper, we name the related model AttCNN and consider applying it to task C2.

The computation of the attention weights in AttCNN model is as follows. Let $X \in \mathbf{R}^t$ be an in-

[1] https://github.com/nicolay-r/SemEval2023-6C-cjp-explanation-with-attention

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 270–274

put document size of t words. Given a set of filters F size of f and CNN convolution $\mathbf{C} \in \mathbf{R}^{t \times f}$, we calculate weights according to the formula (Zhou et al., 2016):

$$\alpha = softmax(w^T \cdot tanh(\mathbf{C}))$$

where $w \in \mathbf{R}^t$ is a hidden vector representation, and $\alpha \in \mathbf{R}^t$ is a normalized attention weights. The application of the attention weights is performed toward the convolved information and results in a modified convolutional matrix $\mathbf{C}_{\text{ATT}} \in \mathbf{R}^{t \times f}$:

$$\mathbf{C}_{\text{ATT}} = Diag(\alpha) \cdot \mathbf{C}$$

Next, given normalized attention weights ($\alpha \in \mathbf{R}^t$), we compose the explanation by applying the following operations:

1. Split original sequence of input tokens into sentences[2].
2. Calculate and order sentences by their average token weights, placing most attentive first.
3. For each sentence apply sliding window size of m_{CNN} and select the region with the most attentive token weights in average.
4. Print each sentence part obtained from the prior step and stop performing so once we reach the limit of explanation of N_{CNN} terms.

1.2 Language Models

We consider the following transformer-based models: RoBERTa (Liu et al., 2019), a robustly pretrained version of the replicated original BERT implementation (Devlin et al., 2019), LegalBERT (Chalkidis et al., 2020), and Longformer (Beltagy et al., 2020). LegalBERT represents a domain-oriented version of the BERT by being pretrained on legal documents. Longformer is a decoder-based transformer model proposed to address the computational complexity problem of the original transformers with an increased amount of input tokens.

A fully-connected layer is added on top of each model to predict the label from the `[CLS]` token embedding. We adopt a self-attention mechanism for explanations as follows:

1. Extract the attention weights of the last head of the top layer toward the `[CLS]` token.
2. Select top-k_{LM} tokens with the highest attention weights.
3. For each selected token, extract the text segment (size of m_{LM}) containing this token, with $m_{\text{LM}}/2$ tokens before and after the selected.

[2]We use NLTK library for annotation (Bird et al., 2009)

"Accepted"	"Rejected"
perjured, redrafted, metamorphosed, edged, swerved, handcuffed, chanted, surveyed, forsaken, detracted, ...	deterred, seconded, plucked, folded, denounced, ante-dating, misbehaved, intrusted, roped, negated ...

Table 1: The most semantically oriented verbs for "Accepted" and "Rejected" classes

2 Resources

We adopt only the datasets provided by organizers: $\text{ILDC}_{\text{single-train}}$ with 5082 documents and $\text{ILDC}_{\text{valid}}$ of 994 documents (Malik et al., 2021). Section 4 provides all the details of these datasets. Organizers mention the noise of the original texts, so we additionally *glew* those word couples that are separated by «− » and could be found in the manually composed list of words[3].

3 Document Processing

We consequently apply our text-processing mechanism and experiment with its stage separately.

v1. Our initial assumption was that legal documents may follow similar structuring templates that may result in the presence of repetitive patterns across the documents. In terms of patterns, we focused on document sentences[4]. Our following assumption was that the presence of similar sentences across all the classes is not in the interest of the adopted classification model. For gaining differences between classes (Günal, 2012), we eliminate similar sentences, mentioned in both classes.

v2. Considering documents without repetitive sentences (v1), our next assumption is as follows: the summary information is likely to appear in the end part of each document. We split every document into a list of sentences in order to reverse the order of the sentences for being used as input afterward. Next, we assess sentence importance with respect to its class ("Accepted" or "Rejected") to keep only salient sentences. Since every sentence could be presented as a list of words, peeled from the punctuation signs, the measurement of sentence salience could be based on its

[3]We publish the list of manually selected words in the main repository

[4]We use NLTK library for annotation

words. For the latter we calculate *Pointwise Mutual Information* (PMI) (Turney, 2002) for further determination of the words[5] w semantic orientation as follows: $SO(w) = PMI(w, \text{ACCEPTED}) - PMI(w, \text{REJECTED})$

To avoid mistaken annotations, we select and consider only $K\%$ of the most frequent words found in documents of each class separately. For a given sentence s, the semantic orientation of the s is calculated as a sum of the orientation of its words. Finally, we drop sentences whose semantic orientation equals zero.

4 Datasets and Experimental Setup

The competition has two stages: *development* and *evaluation*. We consider ILDC$_{\text{single-train}}$ for the preliminary model training with a further assessment on ILDC$_{\text{valid}}$ before the evaluation stage described in Section 2. As for the evaluation stage, we use both of these resources in model fine-tuning.

Since ILDC$_{\text{single-train}}$ is the main resource publicly available, we decided to use it for model parameters selection. Figure 1 illustrates the distribution of the ILDC$_{\text{single-train}}$ document lengths in words for the original text and after processing by v1 (Section 3). While peaks of all distributional plots illustrate the significant portion of documents with the length of 1K-3K words, the mean document length of the processed texts is $\approx$ 4K words shifted from the probability plot peaks due to the presence of 7.5% documents longer than 8K words. Removing repetitive sentences from texts (**v1**) reduces the probability peak from 3% to 2% with standard mean deviation increase by 3.8% (Figure 1). We consider application of **v2** processing towards the joined ILDC$_{\text{single-train}}$ and ILDC$_{\text{valid}}$ collections and keep only $K = 75\%$ of the most oriented entries of each class (Section 3). Table 1 lists the 10 most semantically oriented verbs of each class obtained from the ILDC$_{\text{single-train}}$ and ILDC$_{\text{valid}}$ resources of Section 2.

As for the evaluation stage, task organizers provide ILDC$_{\text{test-c1}}$ and ILDC$_{\text{test-c2}}$ with 1500 and 50 documents respectively (Malik et al., 2021). Statistics of the number of words per document of processed texts are illustrated in Table 2. It is worth mentioning that the processing is only important for training, i.e., ILDC$_{\text{single-train}}$ and ILDC$_{\text{valid}}$, for document difference enhancement between classes.

[5]We consider to keep only `VB*` typed words, using NLTK part-of-speech tagger

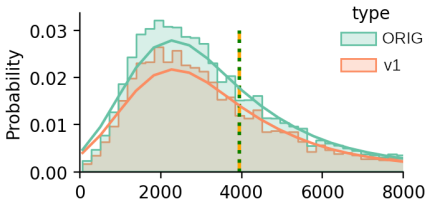

Figure 1: Probability distribution of ILDC$_{\text{single-train}}$ document lengths in words for range [0, 8K] words normalized by the sum of 1 for original texts (ORIG) and v1 (Section 3); vertical dash lines denote expected values

Input ver.	ILDC$_{\text{valid}}$	ILDC$_{\text{test-c1}}$	ILDC$_{\text{test-c2}}$
ORIG$_{\text{words}}$	3752	6419	2315
v1$_{\text{words}}$	3742	6416	2308
v2$_{\text{words, } K\,=\,75\%}$	-	6407	2302

Table 2: Statistics of an average rounded amount of words in the: original texts (ORIG), v1 and v2 (Section 3); application of the processing illustrates a minor reduction < 1% of the contents across all the documents

For CNN and AttCNN models, we consider ILDC$_{\text{single-train}}$ statistics to limit input size by 4K and cover $\approx$ 68% of ILDC$_{\text{single-train}}$ documents completely. We adopt precomputed Word2Vec model from the NLPL repository[6], based on "English Wikipedia Dump of February 2017" with vector size of 300, windows size of 5, and 302,866 word entries. As for additional input features, we consider token position feature size of 5, the sliding window size of 3, and experiment with the number of filters f (Sec. 1.1) of {300, 600, 1200}. We train models with a batch size of 32 documents and terminate this process once the accuracy on ILDC$_{\text{single-train}}$ exceeds 98%. We use AREnets framework (Rusnachenko, 2023) for training and inferring the results. In the case of language models, we consider RoBERTa$_{\text{BASE}}$[7], LegalBERT$_{\text{BASE}}$[8], and Longformer$_{\text{BASE}}$[9] with an attention window size of 512. For the BERT-based models, we consider 510 last tokens[10] (similar to **v2** with $K = 100\%$ in Section 3). As for Longformer, we increase this limit up to 1024 last tokens. All these models were finetuned for 10 epochs with a learning rate of 2e-6 and a batch size of 6.

For the result explanations, in the case of

[6]http://vectors.nlpl.eu/repository/

[7]https://huggingface.co/roberta-base

[8]https://huggingface.co/nlpaueb/legal-bert-base-uncased

[9]https://huggingface.co/allenai/longformer-base-4096

[10]Together with the special tokens `[CLS]` and `[SEP]` (see Section 1.2). Therefore, each input sequence consists of max possible 512 tokens

Model	Input version	C1		C2			
		$\text{ILDC}_{\text{valid}}$	$\text{ILDC}_{\text{test-c1}}$	$\text{ILDC}_{\text{test-c2}}$			
		F1	F1	Rouge1	Rouge2	RougeL	F1
$\text{CNN}_{\text{f=300}}$	v1	63.30	–	–	–	–	–
$\text{CNN}_{\text{f=600}}$	v1	**66.13**	57.16	–	–	–	–
$\text{CNN}_{\text{f=600}}$	v2, $K = 100\%$	–	**60.09**	–	–	–	–
$\text{CNN}_{\text{f=600}}$	v2, $K = 75\%$	–	58.74	–	–	–	–
$\text{CNN}_{\text{f=1200}}$	v1	65.50	–	–	–	–	–
$\text{AttCNN}_{\text{f=600}}$	v1	57.96	–	21.20	4.50	18.10	**68.54**
$\text{AttCNN}_{\text{f=600}}$	v2, $K = 100\%$	–	–	20.89	4.60	17.35	42.58
$\text{AttCNN}_{\text{f=600}}$	v2, $K = 75\%$	–	–	21.46	**4.65**	18.31	47.77
RoBERTa	ORIG	65.81	57.26	22.39	4.52	19.14	50.00
RoBERTa	v1	**66.18**	58.53	22.39	**4.54**	19.04	**52.68**
$\text{Longformer}_{\text{base}}$	ORIG	64.64	–	–	–	–	–
$\text{Longformer}_{\text{base}}$	v1	65.40	55.90	–	–	–	–
$\text{LegalBERT}_{\text{base}}$	v1	65.82	**63.51**	21.33	4.15	18.39	50.00
Final Submission		65.82	**63.51**	21.46	**4.65**	18.31	47.77

Table 3: Results of the single run in tasks C1 and C2; all the language models are BASE sized; gray column highights the results were mentioned in official leaderboard during evaluation stage; the highest results mentioned in leaderboard per every block of models are bolded; K corresponds to the most frequent words selection of **v2** shortening; all the results presented in percents (multiplied by 100)

AttCNN, we select $N_{\text{CNN}}= 320$ and $m_{\text{CNN}}= 100$. In the case of language models, we set the window size $m_{\text{LM}}= 128$ tokens and select top $k_{\text{LM}}= 3$ tokens with the highest attention weights.

5 Result Analysis and Discussion

Table 3 illustrates the results of the applied models with different pre-processing formats in Section 3, with gray-colored columns corresponding to the official leaderboard.

We first analyze the findings of the results in C1. In the case of CNN model, we consider the word window size of 3, and experiment with the various amount of filters $f \in [300, 600, 1200]$. with $f = 600$ selected for the $\text{ILDC}_{\text{test-c1}}$ submissions. The inversion of the text documents and reduction of non-salient sentences (v2) allows an improvement of the past result with extra ≈3-4%. In the case of the language models, our preliminary experiments on $\text{ILDC}_{\text{valid}}$ dataset with RoBERTa illustrate the highest results once using the last 510 tokens as input. Excluding the repetitive sentences from documents (**v1**) improves the results by 0.5%. The best result achieved by RoBERTa on $\text{ILDC}_{\text{test-c1}}$ goes alongside with CNN model with 58.53 by F1. Due to the legal domain of documents adopted in LegalBERT pretraining, this model improves RoBERTa classification results by $\approx 8\%$ with **F1=63.51**, ranked as #8 out of 11 participants. Comparing the result F1 difference with the best submissions, the 3rd best submission (`uottawa.NLP23` team) improves this result by +4.31; by +8.77 with the 2nd best result (`IRIT_IRIS_(C/A)` team), and by +11.34 with the top result (`bluesky` team).

As for the explanation problem C2, in the case of AttCNN (Section 1.1), the submission based on removed repetitive sentences (**v1**) described in Section 3 results in 4.50 by R2. The further omission of sentences with zero semantic orientation (**v2**) allows a slight improvement of R2 by 3%, which is 4.65 by R2. According to the technical log evaluation of other parameters, R1 and RL were slightly better too. In the case of language models, the application of RoBERTa goes alongside with AttCNN approach in terms of R2 and outperforms it in terms of R1 and RL by $\approx 4\%$. Organizers additionally display F1 for model comparisons in C2. Due to the relatively small amount of documents in $\text{ILDC}_{\text{test-c2}}$, we experienced a significant variation in F1-measure results across all the submissions (Table 3, last column). AttCNN illustrates a higher variation of the results with the highest F1=68.54, while language model approaches reach F1 ≈ 50.0-52.8. The final leaderboard reveals only Rouge2 results, according to which all teams have relatively similar results in the range of 4.06-4.73%.

6 Conclusion

In this paper, we provide extensive experiments on attention-based approaches applied to a couple of legal document processing tasks: 1) judgment prediction and 2) explanation of the obtained results. The first task is considered as a text classification

problem, while the second one is an extractive text summarization with a salient sentence selection approach based on the text parts of the most attentive words. Since documents are relatively long and, to the best of our knowledge, could not be completely considered as input of most approaches, we experiment with additional techniques of reduction and mimicking the output class. The findings of our experiments illustrate that treating document endings as input in the models at first results in a 2.1% improvement across all the models. Additional content peeling from non-informative sentences allows us to improve explanation performance by 4% in the case of an attention-based CNN model.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv e-prints*, pages arXiv–2004.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

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.

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.

Serkan Günal. 2012. Hybrid feature selection for text classification. *Turkish Journal of Electrical Engineering and Computer Science*, 20(Sup. 2):1296–1311.

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.

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, Shubham Kumar Nigam, Kripabandhu Ghosh, Shouvik Kumar Guha, Arnab Bhattacharya, and Ashutosh Modi. 2021. 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 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).

Nicolay Rusnachenko. 2023. AREnets: Tensorflow-based framework of attentive neural-network models for text classfication and relation extraction tasks.

Yatian Shen and Xuanjing Huang. 2016. Attention-based convolutional neural network for semantic relation extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2526–2536, Osaka, Japan. The COLING 2016 Organizing Committee.

Peter D. Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 417–424, USA. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762, Lisbon, Portugal. Association for Computational Linguistics.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212, Berlin, Germany. Association for Computational Linguistics.