Information Dynamics and Applications
https://www.acadlore.com/journals/IDA

Extraction of Judgment Elements from Legal Instruments Using an Attention Mechanism-Based RCNN Fusion Model

Jin Ren

School of Information Science and Technology, North China University of Technology, 100144 Beijing, China

* Correspondence: J. Ren (rj@ncut.edu.cn)

Received: 10-14-2024 **Revised:** 11-15-2024 **Accepted:** 11-27-2024

Citation: J. Ren, "Extraction of judgment elements from legal instruments using an attention mechanism-based RCNN fusion model," *Inf. Dyn. Appl.*, vol. 3, no. 4, pp. 223–233, 2024. https://doi.org/10.56578/ida030402.

Abstract: In the field of jurisprudence, judgment element extraction has become a crucial aspect of legal judgment prediction research. The introduction of pre-trained language models has provided significant momentum for the advancement of Natural Language Processing (NLP) technologies, with the Bidirectional Encoder Representations from Transformers (BERT) model being particularly notable for its ability to enhance semantic understanding in unsupervised learning. A fusion model combining BERT and an attention mechanism-based Recurrent Convolutional Neural Network (RCNN) was utilized in this study for multi-label classification tasks, aiming to further extract contextual features from legal texts. The dataset used in this research was derived from the "China Legal Research Cup" judgment element extraction competition, which includes three types of cases (divorce, labor, and lending disputes), with each case type divided into 20 label categories. Four comparative experiments were conducted to investigate the optimization of the model by placing the attention mechanism at different positions. At the same time, previous models were learned and studied and their advantages were analyzed. The results obtained from replicating and optimizing those previous models demonstrate promising legal instrument classification performance.

Keywords: Legal instruments; NLP; BERT; RCNN; Element extraction model

1 Introduction

Currently, big data and artificial intelligence (AI) technologies are playing significant roles across various industries, attracting widespread attention. Their development and application in the judicial field have become key tasks in the future of legal work. The effective use of AI in the judicial domain can bring more convenient smart judicial services to judges, lawyers, and the general public. Smart judicial services primarily involve the application of NLP technology, a core component of AI, to address real-world legal needs. First, it can provide judges and legal professionals with higher work efficiency by extracting key points from complex and lengthy legal instruments and predicting judgment outcomes, thereby assisting experts in decision-making. Second, it can offer high-quality consultation services to the general public, who may not understand legal terms or procedures, by providing specialized advice [1–3]. To promote the development of NLP technology within the smart judiciary, the China Judicial Big Data Research Institute, in collaboration with the Information Center of the Supreme People's Court, organized the "Legal Research Cup" Judicial AI Challenge. This competition used real legal cases from China Judgements Online as a dataset and conducted separate evaluations for tasks such as sentence prediction, crime amount element extraction, and dispute focus element extraction. The main body of a legal instrument typically includes case facts, reasoning (requests), and opinions (decisions). Among these, the case facts are the core of the document, containing the causes, progress, financial loss amounts, and the extent of injuries sustained by participants involved in the case. These factors play an important role in the final judgment and are considered key elements in the judicial decision-making process [4–6]. In most cases, sentences describing the case facts tend to be lengthy and require considerable time to understand. Therefore, research into judgment element extraction in legal instruments is highly significant.

Event extraction refers to the process of presenting unstructured text containing event-related information in a structured format. It has widespread applications in fields such as automatic summarization, question answering, and information retrieval. The goal of event element extraction in the legal domain is to rapidly identify the key elements of events within large volumes of text. For example, in legal instruments related to divorce cases, important

details such as whether the couple has children, the custody arrangement, and whether either party has committed infidelity—factors that significantly influence a judge's final decision—can be quickly identified. Event element extraction is a deeper level of research within the field of information extraction, involving methods and technologies such as deep learning, NLP, and pattern matching.

In recent years, event extraction has attracted considerable attention from research institutions and scholars. Research on event element extraction for English texts has been relatively advanced, with more mature techniques, while the technology for Chinese text remains relatively underdeveloped. In both domestic and international research on event extraction, much of the work is based on the Automatic Content Extraction (ACE) conference and related evaluation corpora. From a technical standpoint, the mainstream approach currently is machine learning, which is preferred over pattern-based methods due to its ease of implementation and strong scalability, making it adaptable to many other fields. Early machine learning methods were based on vocabulary and context features for classification. While in recent years, the use of neural networks for event extraction has become more common. These models require the construction of a robust architecture, after which feature learning occurs autonomously [7–9].

BERT is a pre-trained language representation model. It emphasizes the shift away from traditional unidirectional language models or shallow concatenation of two unidirectional models for pre-training, instead adopting a new Masked Language Model (MLM) approach to generate deep bidirectional language representations. Additionally, the Next Sentence Prediction (NSP) model was used to train the model's ability to understand the relationship between sentences. The BERT paper reported new state-of-the-art results on 11 NLP tasks, which was a remarkable achievement [9–11]. This study primarily focuses on integrating the BERT pre-trained language model with an RCNN model and attention mechanisms, formulating the task of judgment element extraction as a multi-label classification problem. The element extraction dataset from the China "Legal Research Cup" Judicial AI Challenge was used as the dataset in this study. The judgment element sentence extraction task was formulated as a multi-label classification model for factual description sentences. An existing open-source code capable of multi-label classification was selected for study and reproduction, with the goal of understanding, learning, and analyzing methods and implementations of multi-label classification tasks. Through the accumulation of knowledge related to NLP, efforts were made to optimize the model and further explore the BERT model and its structure [12–14].

2 Model Development

2.1 BERT Model

The BERT model is an autoencoding language model. Unlike other language models, BERT is designed to perform pre-training on unlabeled text, where both left and right context from all layers are jointly adjusted to generate deep bidirectional representations. By training on large amounts of unlabeled text, BERT significantly improves the accuracy of NLP tasks [15–17].

BERT not only fully leverages large-scale, unlabeled text to mine rich semantic information but also deepens the NLP model's depth. The BERT model uses the Transformer model as the core algorithmic framework. The Transformer is particularly effective at capturing bidirectional relationships within sentences, meaning it can effectively capture contextual information within the text. This ability is fundamentally based on the attention mechanism.

To enable BERT to handle various downstream tasks, the input sequence was tokenized. In addition to word identifiers, a special classification token [CLS] was inserted at the beginning of each input sequence, and a specific separator token [SEP] was inserted between two input sentences, with two sentences fixed as a sequence in each input. For a given token, its input representation was constructed by adding together the corresponding token, segment, and position embeddings, as shown in Figure 1.

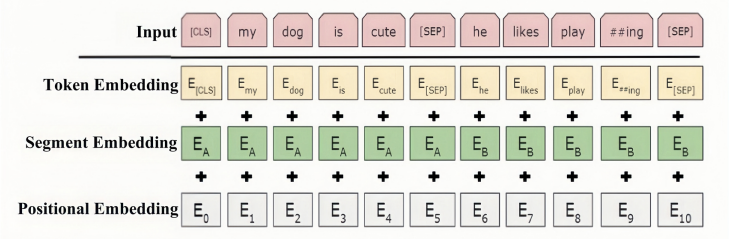

Figure 1. Representation of the BERT input sequence

2.2 Attention Mechanism

When a lengthy legal text is presented, humans tend to focus on the most important information during the reading process. For instance, when encountering a new name, rather than remembering the specific name, attention

is typically directed towards the relationship of that person to other individuals. To enable machines to exhibit similar attention capabilities, the attention mechanism was proposed. The core idea is to focus on the most important parts of the input sequence, that is, to distinguish the impact of different sections of the input on the output. This mechanism, by providing a direct path between the output and input, helps mitigate the vanishing gradient problem [18–20].

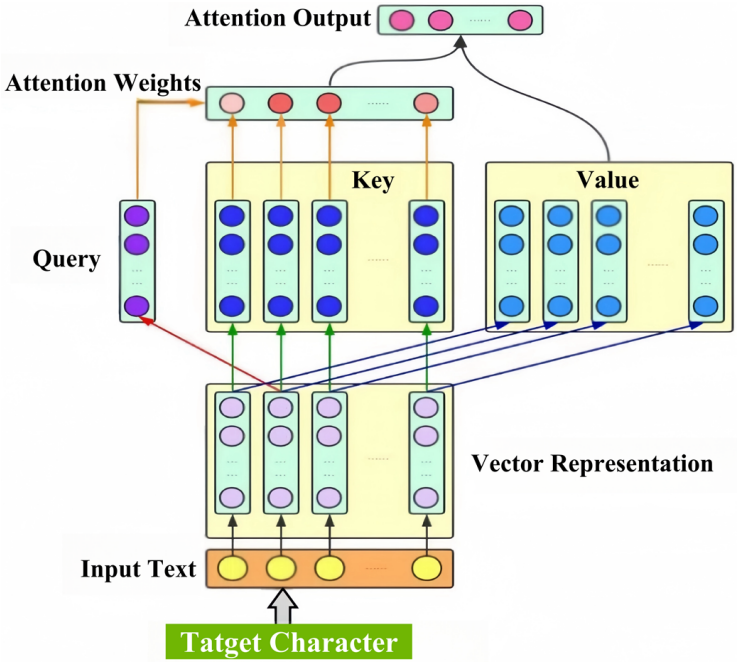

Figure 2. Block diagram of the attention mechanism calculation

The attention mechanism primarily involves three concepts: query, key, and value. In a given text, the target word and its surrounding context words each have their initial value. The attention mechanism treats the target word as the query, while the context words are treated as keys. The similarity between the query and each key is used as a weight, and the value of the context words is integrated into the initial value of the target word. As shown in Figure 2, the attention mechanism takes the semantic vector representations of the target word and the context words as input. First, linear transformations are applied to obtain the query vector representation of the target word, the key vector representations of the context words, and the original value representations of the target word and the context words. Then, the similarity between the query vector and each key vector is computed to determine the weights. These weights are used to compute a weighted fusion of the value vector of the target word and the value vectors of the context words, producing the output of the attention mechanism, i.e., the enhanced semantic vector representation of the target word.

2.3 RCNN Model

The RCNN integrates the structure of Recurrent Neural Network (RNN) with a max-pooling layer, combining the advantages of both Convolutional Neural Network (CNN) and RNN [21–23].

The RCNN model is a bidirectional recurrent structure, which, compared to traditional window-based neural networks, significantly reduces noise and better captures contextual information. It retains a wider range of word order during the learning of text representations. A max-pooling layer, which can automatically identify key features that play a critical role in text classification, is used to capture essential information from the text. The RCNN framework, as shown in Figure 3, consists of three parts: the first part is a bidirectional Long Short-Term Memory (BiLSTM) structure primarily used to learn word representations, the second part is a max-pooling layer, and the third part is a fully connected layer used for learning text representations.

The overall process of constructing the RCNN model is as follows: a) Contextual information is obtained using the BiLSTM, similar to a language model. b) The hidden layer output from the BiLSTM is concatenated with the word vectors. c) The concatenated vector is non-linearly mapped to a lower-dimensional space. d) The maximum value from each time step across the sequence is selected at each position in the vector, producing the final feature vector, similar to the max-pooling process. e) A softmax classifier is applied [24–26].

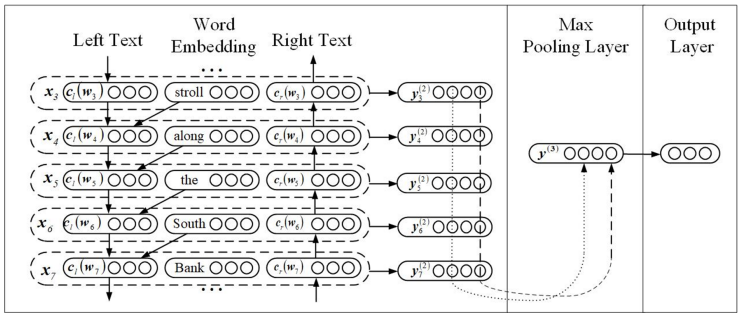

Figure 3. Framework of the RCNN model

3 Multi-label Classification Task Based on the BERT+RCNN+Attention Model

The flowchart of Model 1 is shown in Figure 4. The word embeddings generated by the BERT model were input into the BiLSTM model. The use of a BiLSTM model not only addresses the issues of gradient vanishing and explosion commonly found in traditional RNN models but also mitigates the long-term dependency problems within the network. The specific structure of the BiLSTM is shown in Figure 5, where $x_1, x_2, \ldots x_N$ represent the segmented data, $C_{f,0}, C_{f,1}, \ldots C_{f,N-1}$ are the cell states of the forward Long Short-Term Memory (LSTM) layer, and $C_{b,1}, C_{b,2,\ldots} C_{b,N}$ are the cell states of the backward LSTM layer. Compared to the unidirectional LSTM, the BiLSTM improves the model's performance on sequence classification tasks by extracting deeper semantic information from the text. This allows for better contextual understanding and more accurate inferential decisions. The outputs of the BiLSTM were concatenated in the merging layer to form the word representations. Once all the word representations were computed, they were passed through the max-pooling layer, which converted texts of varying lengths into fixed-length vectors. The max-pooling layer captured the entire text's information. The output was then passed through a fully connected layer, which served as the "classifier" in the network. This layer mapped the "distributed feature representations" to the label space of the samples. The final part of the model was the output layer, where the softmax function was applied to the output vector, processing each of the raw output values. The formula for the softmax function is given in Eq. (1).

$$y_k = e^{a_k} / \left(\Sigma_i e^{a_i}\right) \tag{1}$$

The denominator consolidates all factors from the raw output values, ensuring that the output of the softmax function can be interpreted as "probabilities," with the sum of the outputs equal to 1.

The flowchart of Model 2 is shown in Figure 6. The word embeddings generated by the BERT model were input into the BiLSTM model for word-level representation. These word representations were then concatenated and passed into a hierarchical attention mechanism. The hierarchical attention mechanism assigned different weights to various features in the text, extracting and merging different hierarchical features based on their importance. The output was subsequently passed through a fully connected layer for classification, followed by the final output. Model 2, based on Model 1, replaced the max-pooling layer with a sentence-level attention mechanism, aiming to place greater emphasis on the relationships between sentence-level elements, facilitating a better connection between preceding and succeeding sentences.

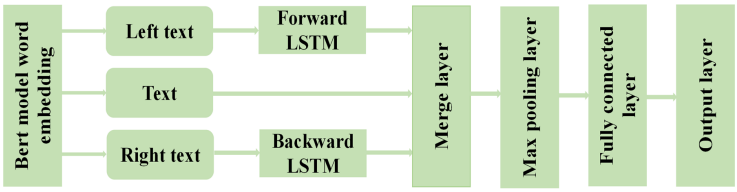

Figure 4. Flowchart of Model 1

The flowchart of Model 3 is shown in Figure 7. The word embeddings generated by the BERT model were input into the BiLSTM model, while an attention mechanism was added to enhance the attention paid to inter-word relationships during word-level representation. This improved the representation of the relationships between words in legal texts. The resulting representations were then concatenated and passed through the max-pooling layer, retaining the most important components for classification. The text was then classified, and the final output was generated. Model 3 built upon Model 1 by adding a word-level attention mechanism after both the forward

and backward LSTM layers, aiming at enhancing the relationships between words by incorporating contextual information.

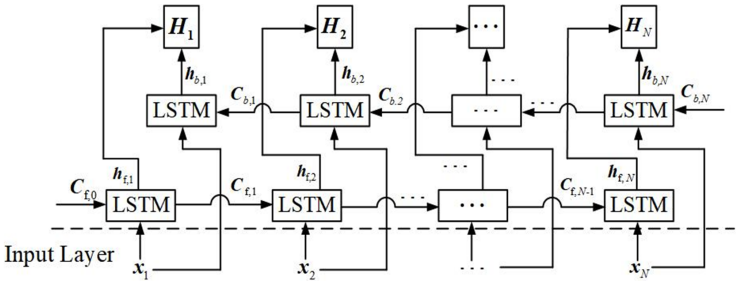

Figure 5. BiLSTM structure

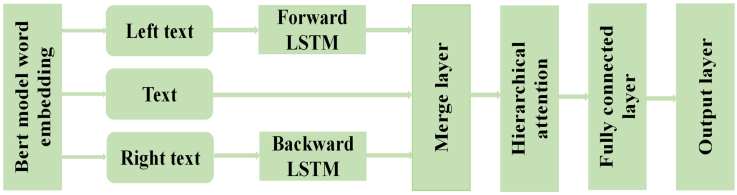

Figure 6. Flowchart of Model 2

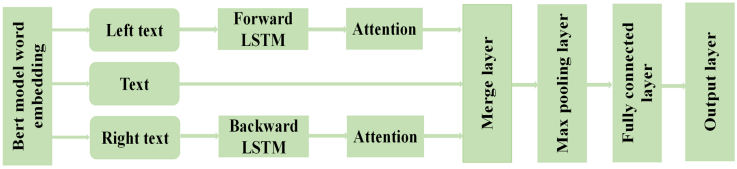

Figure 7. Flowchart of Model 3

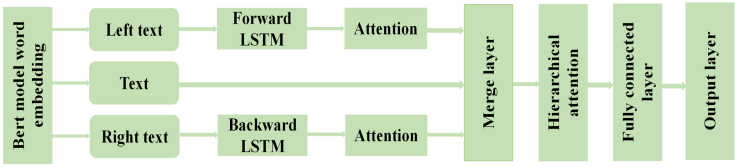

Figure 8. Flowchart of Model 4

The flowchart of Model 4 is shown in Figure 8. The word embeddings generated by the BERT model were input into the BiLSTM model, while an attention mechanism was added to enhance the attention paid to inter-word relationships during the word-level representation process. The resulting word representations were then concatenated and passed into the hierarchical attention mechanism. The hierarchical attention mechanism encoded the weights of the features of each sentence, thereby increasing the relationship between sentences. The output was then sent to the fully connected layer for classification, followed by the final output. Model 4 was based on Model 3, with the max-pooling layer replaced by the sentence-level attention mechanism. Compared to Model 1, this model establishes tighter connections both between sentences and between words within sentences.

4 Experimentation

The model employed in this experiment is a fusion model of BERT+ RCNN+ attention. The platform AutoDL was selected for model construction and execution.

4.1 Dataset

The dataset used in this study was sourced from the "Legal Research Cup" AI Challenge. In the element extraction track, the organizer provided a dataset based on real legal instruments from the China Judgments Online platform

for public evaluation. This dataset contains three types of cases: divorce, labor, and lending disputes. Specifically, the divorce dataset includes 11,685 legal texts, the labor dispute dataset contains 5,680 texts, and the lending dispute dataset consists of 5,123 texts.

Table 1. Multi-label categories for divorce cases

Label	Element Description
DV1	Children born during the marriage
DV2	Custody of children with limited capacity
DV3	Shared marital property
DV4	Payment of child support
DV5	Division of real property
DV6	Separation after marriage
DV7	Second lawsuit for divorce
DV8	Monthly payment of child support
DV9	Divorce granted
DV10	Shared marital debts
DV11	Personal property acquired before marriage
DV12	Statutory grounds for divorce
DV13	Failure to fulfill family obligations
DV14	Existence of a non-marital child
DV15	Provision of appropriate assistance
DV16	Failure to perform divorce agreement
DV17	Damages compensation
DV18	Separation for more than two years due to
DV19	Childrencen living with the non-custodial parent
DV20	Personal property acquired after marriage

Table 2. Multi-label categories for labor cases

Label	Element Description
LB1	Termination of employment relationship
LB2	Payment of wages
LB3	Payment of economic compensation
LB4	Non-payment of full wages
LB5	Existence of an employment relationship
LB6	Failure to sign a labor contract
LB7	Signing of a labor contract
LB8	Payment of overtime wages
LB9	Payment of double wages for not signing a labor contract
LB10	Payment of work-related injury compensation
LB11	Lack of prior labor arbitration procedures
LB12	No requirement to pay compensation for illegal termination of employment
LB13	Economic layoffs
LB14	Non-payment of bonuses
LB15	Illegal collection of property from employees
LB16	Special occupations
LB17	Payment of compensation for work-related death (including funeral allowance and bereavement compensation)
LB18	Employer's early notice of termination
LB19	Legal entity status has been terminated
LB20	Existence of a mediation agreement

The content of legal instruments primarily consists of information related to the individuals and relationships involved in the case, the cause and process of the events, the losses incurred, relevant legal provisions, and the final judgment. The content that significantly influences the judgment result is considered a key reference. In lengthy legal instruments, extracting such important references is essential, which can swiftly assist legal professionals in understanding the specifics of a case and making predictions [27].

In the legal instrument dataset, data were annotated using the format {"labels": [DV1], "sentence": "xxx."}. For example, in the divorce case dataset, a legal text may include the following sentence: {"labels": [DV1], "sentence":

"In February 1998, a daughter named Li Mouyi was born, and on April 15, 2005, a second daughter named Li Moucheng was born. In November 2007, a third daughter named Li Mouding was born."}, indicating that this sentence is labeled as a divorce case (DV1), with children born during the marriage. Similarly, {"labels": [DV3], "sentence": "4. The defendant stated that a Swiss watch given to the plaintiff before marriage is to be considered as belonging to the plaintiff, and the plaintiff will compensate the defendant with 8,500 yuan."} indicates that this sentence is labeled as a divorce case (DV3), involving shared marital property. Furthermore, {"labels": [DV1, DV19, DV2], "sentence": "In September 2012, the daughter Zhao Mouyi moved in to live with the plaintiff Lin Mou. Various expenses for the child's school, living expenses, medical expenses, etc., were all borne by the plaintiff."} indicates a multi-label case, where the sentence is tagged with DV1 (children born during the marriage), DV2 (custody of children with limited capacity), and DV19 (children living with the non-custodial parent).

For different types of cases, the factors that ultimately influence the judgment differ. In this dataset, each type of case was divided into 20 labels based on the key content that impacts the judgment. The labels for divorce cases are shown in Table 1, those for labor cases in Table 2, and those for lending cases in Table 3.

Table 3. Multi-label categories for lending cases

Label	Element Description
LN1	Transfer of creditor's rights
LN2	Loan amount (in ten thousand yuan)
LN3	Existence of a loan agreement
LN4	Lender is a financial institution
LN5	Demand for repayment of principal debt
LN6	Loan by a company, unit, or organization
LN7	Joint and several guarantee liability
LN8	Demand for repayment
LN9	Payment of interest
LN10	Signing of a guarantee agreement
LN11	Existence of a written repayment commitment
LN12	Guarantee agreement is invalid, revoked, or terminated
LN13	Refusal to perform repayment
LN14	Exemption of guarantor from liability
LN15	Guarantor does not bear liability
LN16	Pledgor is a company
LN17	Lender fails to provide loan in accordance with
LN18	agreed date or amount
LN19	Debtor transfers debt
LN20	Agreed interest rate is unclear

4.2 Experimental Setup

The modeling in this experiment was performed using the TensorFlow deep learning framework, with the GPU set to an RTX 3080 (10 GB) and a memory size of 40 GB. The programming language used was Python 3.8, and the network setup and execution were conducted on a cloud server.

The pre-trained BERT model used in this experiment was Google's open-source bert_base_chinese. The hyper-parameters of BERT generally consist of three components: the number of encoder layers in the Transformer (L), the output layer dimension of the model (H), and the number of attention heads in the multi-head attention mechanism (A). Two parameter configurations are provided for BERT models: BERTbase (with parameters L = 12, H = 768, A = 12) and BERTlarge (with parameters L = 24, A = 1024, H = 16) [20]. In this experiment, the BERTbase model parameters were selected.

The main parameters set for different case types in the experiment are listed in Table 4. In this study, *Train_epochs* refers to the number of iterations over the dataset, *Batch_size* indicates the number of samples in each training batch (with weights being updated after each training pass through backpropagation), *Max_length* specifies the maximum allowable length of input text strings in the input field, and Learning_rate represents the initial learning rate.

The loss function used in this experiment was the Focal loss method. This function reduces the weight of easily classified samples, allowing the model to focus more on the difficult-to-classify samples during training. The expression for the Focal loss function is shown in Eq. (2):

$$F_l\left(p_t\right) = -\mathrm{a} * \left(1 - p_t\right)^{\gamma} * \log\left(p_t\right) \tag{2}$$

where, p_t represents the predicted probability of the sample, a is the sample weight, r is the modulation factor. When $r = 0$, the Focal loss function is equivalent to the standard cross-entropy loss function. When $r > 0$, the Focal loss function effectively mitigates the issue of class imbalance. The parameters selected for this experiment were $a = 0.25$ and $r = 2$.

Table 4. Parameter settings

Case Type	*Train_Epochs*	*Batch_Size*	*Max_Length*	*Learning_Rate*
Divorce	10	20	128	$2e-5$
Labor	10	20	150	$2e-5$
Lending	10	10	200	$2e-5$

4.3 Evaluation Metrics and Results

For conventional classification model evaluation, precision, recall, and the F1-score are commonly used as metrics. In this experiment, the F1-score was selected as the evaluation criterion. The F1-score is the harmonic mean of precision and recall. Considering only precision or only recall would not provide a comprehensive measure of a model's performance; thus, the F1-score was used to balance both precision and recall. The experimental results are shown in Table 5, Table 6, and Table 7.

Table 5. Data results of the divorce cases

Model	F1-score
1	0.768
2	0.770
3	0.758
4	0.764

Table 6. Data results of the labor cases

Model	F1-score
1	0.672
2	0.680
3	0.660
4	0.679

Table 7. Data results of the lending cases

Model	F1-score
1	0.666
2	0.687
3	0.658
4	0.686

From the three tables above, it can be observed that Model 2 consistently achieved the highest F1-score. Model 3 exhibited a lower F1-score compared to Model 1, and similarly, Model 4 showed a lower F1-score than Model 2. This suggests that the inclusion of word-level attention mechanisms did not yield positive results and, in fact, led to a decrease in the F1-score. When comparing Model 2 to Model 1, the F1-score was improved, indicating that replacing the max-pooling layer with a hierarchical attention mechanism could optimize the model.

The term loss refers to the loss value of the training set. Figure 9, Figure 10, and Figure 11 display the loss curves for each model in the divorce, labor, and lending cases, respectively.

4.4 Analysis of the Experimental Results

The data presented above indicate that the fusion model of BERT combined with RCNN is capable of effectively performing multi-label classification tasks. As shown in Table 5, Table 6, and Table 7, replacing the max pooling

layer with a hierarchical attention mechanism network improves the F1-score across all three case types. This suggests that the hierarchical attention mechanism, compared to the max pooling layer, is able to more accurately focus on the relationships between sentences in legal texts. This is due to the embedding process of the hierarchical attention mechanism, which increases the degree of sentence-level association, whereas the max pooling layer does not emphasize sentence relationships as effectively. However, in the legal text dataset, adding an attention mechanism after the BiLSTM to focus on word-level relationships did not lead to an improvement in the F1-score. Instead, it resulted in a slight decrease, indicating that the word-level attention mechanism increased the complexity of the model without optimizing it. The loss curves shown in Figure 9, Figure 10, and Figure 11 reveal that the loss for Model 1 and Model 3 decreases more slowly within the first 200 steps, while for Models 2 and 4, where the max pooling layer is replaced with an attention mechanism, the loss decreases more rapidly. For all models, as the number of neural network training steps increases, the loss decreases to below 0.3 and stabilizes. Overall, Model 2, which is the BERT-RCNN fusion model with the max pooling layer replaced by the hierarchical attention mechanism, performs better in classification. It rapidly reduces the loss and stabilizes as training progresses. The replacement of the max pooling layer with the hierarchical attention mechanism effectively captures the semantic relationships between sentences, concentrating the learning on the sentences most relevant to text classification. This results in further optimization of the network structure and a significant improvement in classification performance.

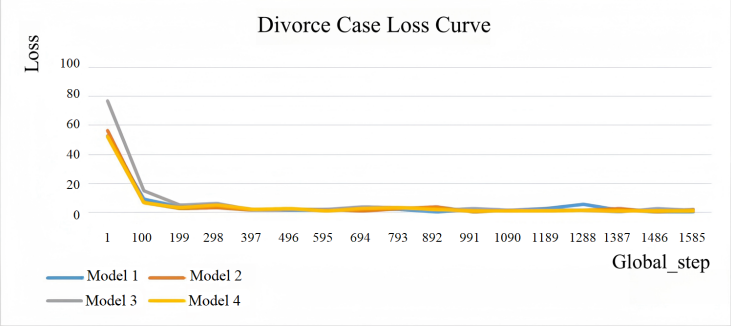

Figure 9. Loss curve for the divorce cases

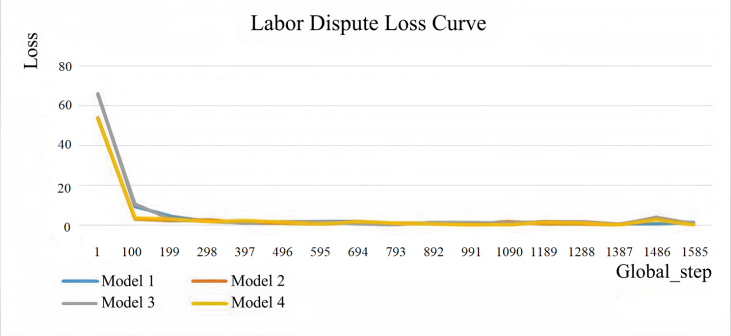

Figure 10. Loss curve for the labor cases

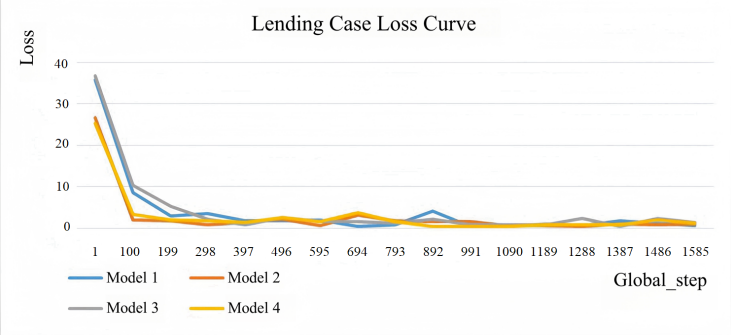

Figure 11. Loss curve for the lending cases

5 Conclusion

In this study, the BERT pre-trained model was combined with the improved traditional RCNN model for multi-label classification tasks of legal instruments. The BERT model is capable of extracting key features of words within sentences and extracting relational features at multiple levels, thereby providing a more comprehensive representation of sentence semantics. By integrating it with the RCNN model augmented with an attention mechanism, the degree of inter-sentence relationships in legal texts was increased during the embedding process, enabling concurrent execution. Compared to traditional models, this algorithm shows improved classification accuracy. This is due to the accumulation of training data from millions of iterations within the pre-trained model, which allows the word embedding process in the proposed method to retain more contextual and syntactic information. The inclusion of a hierarchical attention mechanism in the model enables greater focus on the relationships between key sentences during the learning process, thereby achieving more accurate classification results.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares no conflict of interest.

References

[1] H. Zhang, B. Z. Pan, and Y. Zhang, "Judgment elements extraction for factual description of legal documents based on deep learning," *Comput. Appl. Softw.*, vol. 38, no. 9, pp. 160–166, 2021.

[2] J. H. Dai, R. Y. Peng, L. Xu, C. Jiang, D. J. Zeng, and Y. D. Li, "A survey of information extraction based on deep neural networks," *J. Southwest China Norm. Univ. (Nat. Sci. Ed.)*, vol. 47, no. 4, pp. 1–11, 2022. https://doi.org/10.13718/j.cnki.xsxb.2022.04.001

[3] S. Paul, A. Mandal, P. Goyal, and S. Ghosh, "Pre-trained language models for the legal domain: A case study on Indian law," in *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*, 2023, pp. 187–196. https://doi.org/10.1145/3594536.3595165

[4] V. Naik, P. Patel, and R. Kannan, "Legal entity extraction: An experimental study of NER approach for legal documents," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 3, 2023. https://doi.org/10.14569/IJACSA.2023.0140389

[5] A. Sleimi, N. Sannier, M. Sabetzadeh, L. Briand, and J. Dann, "Automated extraction of semantic legal metadata using natural language processing," in *2018 IEEE 26th International Requirements Engineering Conference (RE), Banff, AB, Canada*, 2018, pp. 124–135. https://doi.org/10.1109/RE.2018.00022

[6] E. Hammami, R. Faiz, and I. Akermi, "A dynamic convolutional neural network approach for legal text classification," in *International Conference on Information and Knowledge Systems*. Springer, Cham, 2021, pp. 71–84. https://doi.org/10.1007/978-3-030-85977-0_6

[7] Y. Fang, X. Tian, H. Wu, S. Y. Gu, Z. Wang, F. Wang, J. L. Li, and Y. Weng, "Few-shot learning for Chinese legal controversial issues classification," *IEEE Access*, vol. 8, pp. 75 022–75 034, 2020. https://doi.org/10.1109/ACCESS.2020.2988493

[8] X. X. Yang, Z. F. Wang, Q. Wang, K. Wei, K. Q. Zhang, and J. G. Shi, "Large language models for automated Q&A involving legal documents: A survey on algorithms, frameworks and applications," *Int. J. Web Inf. Syst.*, vol. 20, no. 4, pp. 413–435, 2024. https://doi.org/10.1108/IJWIS-12-2023-0256

[9] J. Chen, "Research on long text classification based on the combination of BERT feature representation and attention mechanism," *Comput. Era*, no. 5, pp. 136–139, 144, 2023. https://qikan.cqvip.com/Qikan/Article/Detail?id=7109527681

[10] Y. G. Lyu, Z. H. Wang, Z. C. Ren, P. J. Ren, Z. M. Chen, X. Z. Liu, Y. J. Li, H. S. Li, and H. Y. Song, "Improving legal judgment prediction through reinforced criminal element extraction," *Inf. Process. Manag.*, vol. 59, no. 1, p. 102780, 2022. https://doi.org/10.1016/j.ipm.2021.102780

[11] Q. H. Zhao, T. H. Gao, and N. Guo, "LA-MGFM: A legal judgment prediction method via sememe-enhanced graph neural networks and multi-graph fusion mechanism," *Inf. Process. Manag.*, vol. 60, no. 5, p. 103455, 2023. https://doi.org/10.1016/j.ipm.2023.103455

[12] G. Y. Feng, Y. B. Qin, R. Z. Huang, and Y. P. Chen, "Criminal Action Graph: A semantic representation model of judgement documents for legal charge prediction," *Inf. Process. Manag.*, vol. 60, no. 5, p. 103421, 2023. https://doi.org/10.1016/j.ipm.2023.103421

[13] Z. Q. Lin, F. Yang, X. Y. Wu, J. S. Su, and X. Y. Wang, "A Feedback-Enhanced Two-Stage Framework for judicial Machine Reading Comprehension," *Eng. Appl. Artif. Intell.*, vol. 123, p. 106178, 2023. https://doi.org/10.1016/j.engappai.2023.106178

[14] B. F. Fang, C. M. Zheng, H. Wang, and T. T. Yu, "Two-stream fused fuzzy deep neural network for multiagent learning," *IEEE Trans. Fuzzy Syst.*, vol. 31, no. 2, pp. 511–520, 2023. https://doi.org/10.1109/TFUZZ.2022.3214001

[15] W. G. Wang, Y. W. Chen, H. Cai, Y. N. Zeng, and H. Y. Yang, "Judicial document intellectual processing using hybrid deep neural networks," *J. Tsinghua Univ. (Sci. Technol.)*, vol. 59, no. 7, pp. 505–511, 2019.

[16] H. Zhang, Z. C. Dou, Y. T. Zhu, and J. R. Wen, "Contrastive learning for legal judgment prediction," *ACM Trans. Inf. Syst.*, vol. 41, no. 4, pp. 1–25, 2023. https://doi.org/10.1145/3580489

[17] A. Pal, S. Rajanala, R. C. W. Phan, and K. Wong, "Self supervised BERT for legal text classification," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, *Rhodes Island, Greece*, 2023, pp. 1–5. https://doi.org/10.1109/ICASSP49357.2023.10095308

[18] Y. G. Lyu, J. T. Hao, Z. H. Wang, K. Zhao, S. Gao, P. J. Ren, Z. M. Chen, F. Wang, and Z. C. Ren, "Multi-Defendant Legal Judgment Prediction via hierarchical reasoning," in *Findings of the Association for Computational Linguistics: EMNLP 2023*. *Singapore*: Association for Computational Linguistics, 2023, pp. 2198–2209. https://doi.org/10.18653/v1/2023.findings-emnlp.145

[19] S. D. Hu, N. Ding, H. D. Wang, Z. Y. Liu, J. G. Wang, J. Z. Li, W. Wu, and M. S. Sun, "Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*. *Dublin, Ireland*: Association for Computational Linguistics, 2022, pp. 2225–2240. https://doi.org/10.18653/v1/2022.acl-long.158

[20] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "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*. *Minneapolis, MN, United States*: Association for Computational Linguistics, 2019, pp. 4171–4186. https://doi.org/10.18653/v1/n19-1423

[21] A. Adhikari, A. Ram, R. Tang, W. L. Hamilton, and J. Lin, "Exploring the limits of simple learners in knowledge distillation for document classification with DocBERT," in *Proceedings of the 5th Workshop on Representation Learning for NLP*. Association for Computational Linguistics, 2020, pp. 72–77. https://doi.org/10.18653/v1/2020.repl4nlp-1.10

[22] P. M. Kien, H. T. Nguyen, N. X. Bach, V. Tran, M. Le Nguyen, and T. M. Phuong, "Answering legal questions by learning neural attentive text representation," in *Proceedings of 28th International Conference on Computational Linguistics*. Association for Computational Linguistics, 2020, pp. 988–998. https://doi.org/10.18653/v1/2020.coling-main.86

[23] L. L. Gan, K. Kuang, Y. Yang, and F. Wu, "Judgment prediction via injecting legal knowledge into neural networks," in *Proceedings of the AAAI conference on artificial intelligence*, 2021, pp. 12 866–12 874. https://doi.org/10.1609/aaai.v35i14.17522

[24] J. Y. Sun, S. B. Huang, and C. Wei, "A BERT-based deontic logic learner," *Inf. Process. Manag.*, vol. 60, no. 4, p. 103374, 2023. https://doi.org/10.1016/j.ipm.2023.103374

[25] W. J. Lu, Y. Duan, and Y. T. Song, "Self-attention-based convolutional neural networks for sentence classification," in *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, *Chengdu, China*, 2020, pp. 2065–2069. https://doi.org/10.1109/ICCC51575.2020.9345092

[26] L. Yue, Q. Liu, B. Jin, H. Wu, K. Zhang, Y. An, M. Cheng, B. Yin, and D. Wu, "NeurJudge: A circumstance-aware neural framework for legal judgment prediction," in *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, 2021, pp. 973–982. https://doi.org/10.1145/3404835.3462826

[27] H. X. Zhong, C. J. Xiao, C. C. Tu, T. Y. Zhang, Z. Y. Liu, and M. S. Sun, "How does NLP benefit legal system: A summary of Legal Artificial Intelligence," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020, pp. 5218–5230. https://doi.org/10.18653/v1/2020.acl-main.466