Research Article

BERT_LF: A Similar Case Retrieval Method Based on Legal Facts

Weifeng Hu1,1 Siwen Zhao1,1 Qiang Zhao1,1 Hao Sun1,1 Xifeng Hu1,1 Rundong Guo1,1 Yujun Li1,1 Yan Cui2 and Long Ma3

1School of Information Science and Engineering, Shandong University, Qingdao 266200, China
2Institute of Sociology, Chinese Academy of Social Sciences, Beijing 100732, China
3Troy University 321D McCall Hall (MSCX), Troy AL 36082, USA

Correspondence should be addressed to Yujun Li; liyujun@sdu.edu.cn and Yan Cui; cuiyanshky@sina.com

Received 23 February 2022; Accepted 30 March 2022; Published 30 April 2022

Academic Editor: Yingjie Wang

Copyright © 2022 Weifeng Hu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the development of smart justice in China, the Supreme People’s Court implements the system of compulsory retrieval for similar cases continuously and comprehensively, more and more judicial big data has been continuously disclosed, and the application of case retrieval is more extensive, and then, the accuracy of similar case search results needs to be urgently improved. Legal case retrieval is a special search task; for a given query case, it refers to the searching for similar cases. Different from traditional text search, legal case retrieval has different characteristics and greater challenges, for its query case is longer and more complex than common keyword queries and short article queries. In addition, the definition of dependencies between query cases and candidate cases differs from general dependencies based on text or topic. In order to solve these problems, we propose a method for similarity case retrieval based on the legal facts, and our model combine the topic distribution and legal entity facts to make the document representation vector more suitable for legal scenarios. At the same time, the method of paragraph aggregation based on BERT is used to encode context semantic information and solve the problem of long text. The experimental results show that our method is superior to the existing method.

1. Introduction

In many legal systems, similar case retrieval is of great significance to ensure legal fairness. With the development of smart justice in China and the increasing number of digitized legal documents, automatic retrieval of legal cases has attracted more and more attention in the research field of information retrieval (IR) [1–3]. In recent years, researchers have made many typical contributions to the retrieval of legal information [4–8].

The purpose of legal case retrieval is to identify cases that are similar to the given case. China has provided a guiding case series that can be referred to for the trial of similar cases. Guiding cases are composed of titles, keywords, key points of adjudication, relevant laws and regulations, basic case facts, adjudication results, adjudication reasons, and notes including the names of the effective adjudication and adjudicators. The problem of studying similar cases is essentially the study of text similarity. However, the legal case retrieval task differs greatly from traditional text retrieval in terms of the length of the case text, the definition of relevance, and the accessibility of the legal dataset. Based on the research of Shao et al. [9], there are several challenges in solving the problem under the existing text similarity method:

Challenge 1. Legal cases are often long texts, which causes models to fail to handle all useful information when establishing vector representations of text. At present, the memory ability of the most commonly used neural network models in the text field, such as LSTM, is not strong, and their application effect in long text is not good, which also leads to the poor effect of the general text similarity model in the legal field. Xiao et al. [10] proposed a model that combines local sliding window attention and global task-driven full attention, called Lawformer, for processing long texts.

Challenge 2. The similarity of legal cases differs from the generic textual similarities and, to some extent, also goes beyond the general definition of subject matter relevance.
[2]. It needs to explore the similarity of the legal facts contained in the legal case text. Traditional text similarity method can indeed learn the semantics similarity, but the model does not understand the knowledge of the legal field, so it may not be able to learn the deeper legal-related logical relationship under the surface semantics, which leads to the failed of finding highly similar legal cases using text similarity methods alone. Therefore, it is crucial to identify the similarities of cases in terms of legal issues and legal processes, which requires a full understanding of the legal case text.

Challenge 3. Collecting large amounts of legal case data, as well as similar case datasets, is a challenge. On the one hand, in many legal systems, the download of large legal documents is restricted. On the other hand, the cost of obtaining accurate correlation judgments is higher due to the need for expertise in the legal field. The lack of data hampers the training process for deep neural models.

What is more, the text structure of legal judgment documents is different from that of an ordinary text. The generic text similarity model mainly considers the structural characteristics of the text, such as syntactic structure, but although the legal judgment instrument is an unstructured text, it often has specific format requirements, so the general text similarity method cannot accurately represent the legal text; if the structural characteristics of the judgment document can be combined with the calculation of the similarity for the general text, it may produce better results.

To solve the problems above, we proposed a BERT-LF model; for challenge 1, literature [9] and literature [11–14] explored the long text problem applied by BERT, respectively, and their work included sentence-level fraction aggregation, paragraph-level fraction aggregation, and paragraph-level representation aggregation, so that the problem has been roughly solved, which inspired us to infer the similarity of the entire legal case by aggregating paragraph-level semantic interactions. For challenge 2, we proposed a legal case representation method based on legal facts, combining with the topic distribution. The deep combination of legal facts, document topic, and semantic information makes the document representation vector more suitable for legal scenarios. Further, we used an attention mechanism to distinguish the importance of legal information between paragraphs. For challenge 3, we crawled the judgment document data from “China Judgements Online” to train the topic model to adapt it to the legal scene. Our experiments were conducted on the legal case retrieval dataset [15], and the results proved the effectiveness of the proposed method.

2. Related Work

In the past researches, a large number of text retrieval models have been proposed, especially for specific texts. Literature [16–18] solved the problems of complex feature dimension and difficult retrieval of text data. Common approaches for early semantic representation include vector spatial models, topic models, and their variants such as the classic LDA [19]. But research in literature [20] showed that the similarity under the same topic still needed to be improved. With the advent of word embedding, information retrieval has now shifted to neural information retrieval. Researchers began using dense vector representations of words and documents based on deep learning models [21–24] as input of machine learning algorithms. Traditional bag-of-words IR models include BM25 [25], TF-IDF [26], and LMIR [27]. Mandal et al. [28] compared the effects of four unsupervised text vector generation models, TF-IDF, word2vec, LDA, and doc2vec, when calculating legal text similarity, and tested it on an Indian dataset containing 47 case pairs; the result showed that doc2vec worked best. Vo et al. [29] also indicated that text semantic representations based on word embedding are helpful in the field of legal text retrieval. Meanwhile, researches in literature [30, 31] showed the effectiveness of neural embedding of texts in legal information retrieval.

In view of the text similarity problem for Chinese, some scholars made a series of improvements to the classical similarity method. Li et al. [32] proposed a text similarity calculation algorithm based on improved VSM, which took into account the influence of the same feature words between similar texts. Huang et al. [33] proposed a supervised-WMD algorithm, which added new document features and movement costs to WMD and solved the problem that the WMD cannot take useful classification information into account.

Further, in the field of Chinese legal case retrieval, Lv and Hou [34] improved topic distribution model and designed a legal case recommendation algorithm. They argued that the words generated by the topic distribution model have different representations of legal texts. Thus, they reduced the probability distribution of words which appeared frequently but carried little weight with the legal text; what is more, they improved the probability distribution of words that did not appear frequently but were helpful for the representation of the legal text. In the similarity module of Xiang [35], the keywords were extracted by natural semantics and TF-IDF, the keywords of the judgment document were formed by semantic and frequency, and then, the judgment document was converted to vectors by the keyword table. Obviously, if you just consider whether the keywords are the same, you will ignore the contextual information. Wang et al. [36] compared the effect of the TF-IDF model, the LDA model, and the improved LLDA model on the task of case similarity and found that the TF-IDF had the worst effect and the LLDA model had the best effect. And they also point out that if you want to get a good effect, the parameters of the topic model need manual intervention. These similarity calculation methods based on word perspectives ignore the meaning of word order and context. Some subsequent studies have used the vector representation of word2vec to calculate case similarity. Deng [37] fused the word2vec, doc2vec, and TF-IDF algorithms and used them in the calculation of case similarity. Li [38] designed a method for calculating the similarity of documents that combined bipartite diagrams and syntactic information. The compressed document content was used to calculate the text similarity, and good results were obtained. Liu et al. [39] proposed a similar case recommendation model based on neural networks, which first used legal facts to
guide the generation of text representation vectors for each case, and then used the generated vectors to calculate the similarity scores of any pair of cases, and the set of cases with the highest similarity was used as the recommended similar cases. Although the above researchers have achieved certain results, most of the models are not designed for long legal documents.

Since BERT [40] has made significant improvements in various NLP tasks and achieved state-of-the-art performance in 11 missions, pretrained language models have attracted a great deal of attention in the field of information retrieval. Recently, several studies have elucidated the application of BERT in legal case retrieval, such as literature [9] and literature [11-14].

3. Materials and Methods

3.1. Task Description. Legal case retrieval task refers to finding cases similar to a given query case in the candidate cases set \[D\]. Formally, given a query case \(q\), and a set of candidate cases \(D = \{d_1, d_2, \ldots, d_n\}\), the task of legal case retrieval is to determine the supporting case \(D^* = \{d_1^*, d_2^*, \ldots, d_n^*\} \in D\) noticed \(d_i^*\), where noticed \((d_i^*, q)\) indicates that \(d_i^*\) is legally similar to the query case \(q\). Both the queries and candidates are long texts containing descriptions of legal facts.

3.2. Architecture Overview. As shown in Figure 1, in general, the entire framework of our model contains three modules, the first part is the legal feature encoding module, which contains the semantic encoding part based on BERT, the topical encoding module based on LDA model, and the encoding part based on legal entities; the second part is the aggregation of encoding; before entering the third part, the second part is responsible for encoding and aggregating the output of the first part; and the third part is the relational computation based on the attention mechanism.

3.3. Legal Feature Encoding. When judging whether two cases are similar, we are actually considering whether the legal facts and the logic of the events contained in the two cases are similar. Therefore, legal fact information is extracted through three coding modules; we capture the semantic context information of the case through the BERT-based module, cluster the topic information by the topical encoding module, and strengthen the role of legal facts information more accurately through the legal entity encoding module. And then, we aggregate all the above encodings to represent the paragraph-level information.

3.4. Semantic Encoding. Drawing on the analysis of the literature [9, 11-14], in the part of semantic encoding, we use paragraph aggregation architecture based on BERT. Firstly, we divide the long text into paragraphs that BERT can handle and then get semantic encoding of the query and candidate paragraphs based on a pretrained BERT model. On the one hand, it can take advantage of BERT’s strong semantic learning ability, and on the other hand, it can solve the problem of long text encoding for legal cases.

Formally described, for a query document \(q\) and candidate document \(d_k\) which can be represented as \(q = (p_{q1}, p_{q2}, \ldots, p_{qN})\) and \(d_k = (p_{k1}, p_{k2}, \ldots, p_{kM})\) where \(N\) and \(M\) denote the total number of paragraphs for \(q\) and \(d_k\), respectively. For each paragraph in \(q\) and \(d_k\), we construct a paragraph pair \((p_{qi}, p_{kj})\), where \(1 \leq i \leq N\) and \(1 \leq j \leq M\), along with the reserved tags (i.e., [CLS] and [SEP]), serve as the input of BERT. We use the pretrained Chinese BERT model which called BERT-Base-Chinese (https://github.com/google-research/bert/blob/master/multilingual.md) to obtain a representation for each passage. Positional embeddings are added to capture word order, and these embeddings are fed into the transformer layers, where each layer of transformers generates a new upper and lower context embedding representation by calculating the weighted sum of the token embeddings. The weight value is calculated by multthead attention mechanism. Words with a large attention weight are considered more relevant to the target word. Different attention matrices capture different types of word relationships, such as exact matching or synonym relationships. Finally, the final hidden layer vector output of the first token [CLS] is represented as the semantic aggregate of query-candidate paragraphs. As shown in Figure 1, we use the output embedding of the first token as the representation for the entire query-candidate pair:

\[
C_{ij} = BERT\left(p_{qi}, p_{kj}\right). \tag{1}
\]

By this way, we can get an interaction matrix of all query-candidate paragraphs \(C\), where the semantic representation of each paragraph for \(p_{qi}\) and \(p_{kj}\) is \(C_{ij}, C_{ij} \in \mathbb{R}^{HB}\). Next, interaction matrix \(C\) is further encoded with GRU model. Then, we get a sequence of hidden states generated by the forward GRU \(h_{qk} = [h_{qk1}, h_{qk2}, \ldots, h_{qkN}], h_{qk} \in \mathbb{R}^{HB}\).

3.5. Topical Encoding. In this section, we obtain the topic probability interaction matrix of the query paragraph and the candidate paragraph pairs according to the inverse process of generating documents in LDA model.

As we all know, the process of generating documents in LDA is document generation, topic generation, and word generation, which are divided into the following five steps:

1. Select a document \(m\) based on the prior probability
2. Sample the topic distribution \(\theta\) of the generated document from the Dirichlet distribution
3. Sample the topic of the \(j\)-th word of the document from the polynomial distribution \(\theta\) of the topic
4. Generate a word distribution corresponding to the topic from the Dirichlet distribution based on prior knowledge \(\varphi\)
5. Sample from the polynomial distribution of words \(\varphi\) to produce the final word \(\omega\)

However, in the process of inferring the distribution of topics in a document, only the word \(w\) in document \(d\) is observable, the subject \(z\) is hidden, and the posterior
The posterior distribution of topics is solved by the Gibbs sampling method.

According to formula (2), the topic distribution of paragraphs $P_{qi}$ and $P_{kj}$ is $ZP_{qi} = [ZP_{qi-1}, ZP_{qi-2}, \ldots, ZP_{qi-v}]$ and $ZP_{kj} = [ZP_{kj-1}, ZP_{kj-2}, \ldots, ZP_{kj-v}]$, respectively. Then, we use a similarity formula (3) to get the similarity interaction matrix between the query paragraphs and the candidate paragraphs about topic probability distribution $T_{qik} = [T_{qik1}, T_{qik2}, \ldots, T_{qikN}]$, where each element is represented by $T_{qik}$.

$$T_{qik} = \frac{ZP_{qi-1} \ast ZP_{kj-1} + ZP_{qi-2} \ast ZP_{kj-2} + \cdots + ZP_{qi-v} \ast ZP_{kj-v}}{\sqrt{(ZP_{qi-1})^2 + (ZP_{qi-2})^2 + \cdots + (ZP_{qi-v})^2} \ast \sqrt{(ZP_{kj-1})^2 + (ZP_{kj-2})^2 + \cdots + (ZP_{kj-v})^2}},$$

where $v$ is a hyperparameter, representing the number of topics.

### 3.6. Legal Entity Encoding

We mainly focus on criminal cases in China. Referring to the previous research [39, 41, 42], this study mainly focuses on several parts of legal case including criminal offence, criminal entity type and compensation behavior, criminal consequences, reconciliation, and criminal charge. These legal entities contain the legal facts that have a decisive influence on the judgment.
Criminal offence: wound, hurt, injury, mutual beatings, intentional injury due to trivial disputes, etc.

Criminal consequences: economic loss, death, minor injury, serious injury, minor injury of the first degree, etc.

Compensation behavior: repentance, meritorious service, voluntary admission of guilt, voluntary surrender, etc.

Reconciliation: reach a settlement agreement, obtain the victim’s forgiveness, etc.

Criminal charge: intentional injury crime

For the given query-candidate paragraph pair, the resulting entity sets are criminal entity type: \(E_{ct-qi}^t\), \(E_{ct-kj}^t\), criminal offence: \(E_{co-qi}^t, E_{co-kj}^t\), criminal consequences: \(E_{ce-qi}^t, E_{ce-kj}^t\), compensation behavior: \(E_{cb-qi}^t, E_{cb-kj}^t\), reconciliation: \(E_{r-qi}^t, E_{r-kj}^t\) and criminal charge: \(E_{cg-qi}^t, E_{cg-kj}^t\). Then, we calculate the similarity of query-candidate paragraph pairs in terms of legal entities.

Firstly, we splice all entities of query paragraphs and candidate paragraphs into two short texts, \(T_{qi}^t\) and \(T_{kj}^t\), in the order of criminal entity type, criminal offence, criminal consequences, compensation behavior, reconciliation, and criminal charge.

And then, we use the pretrained Chinese BERT BERT-Base-Chinese to obtain a representation for \(T_{qi}^t\) and \(T_{kj}^t\), respectively and obtain tensor \(T_{qi}^t\) and tensor \(T_{kj}^t\):

\[
T_{qi}^t = \text{BERT}(T_{qi}), \quad T_{kj}^t = \text{BERT}(T_{kj}).
\]  

Finally, we calculate the cosine similarity of tensor \(T_{qi}^t\) and tensor \(T_{kj}^t\), as the entity similarity value of query-candidate paragraph pair.

\[
\text{sim}_e(E_{qi}^t, E_{kj}^t) = \cos \left( T_{qi}^t, T_{kj}^t \right).
\]  

Then, we can obtain an interaction matrix between the query paragraphs and the candidate paragraphs about legal entity related \(LE_{qk} = [LE_{qk1}, LE_{qk2}, \ldots, LE_{qkN}]\), where each element \(LE_{qk} = [\text{sim}_e(E_{q1}^t, E_{k1}), \text{sim}_e(E_{q1}^t, E_{k2}), \ldots \cdot \text{sim}_e(E_{q1}^t, E_{kN})]\).

3.7. Encoding Aggregation and Similarity Calculation. In this section, semantic encoding, topic distribution encoding, and legal entity encoding are aggregated, and the similarity of query-candidate pairs is calculated as follows:

\[
E_{qk} = \beta_1 h_{qk} + \beta_2 T_{qk}^t + \beta_3 LE_{qk},
\]  

where \(\beta_1, \beta_2,\) and \(\beta_3\) are weight parameters. Then, for each paragraph of the query document, we use max pooling operation to get the strongest matching paragraph in candidate documents, resulting in a sequence vector expressed as \(E_{qk} = [E'_{qk1}, E'_{qk2}, \ldots, E'_{qkN}]\), where \(E'_{qk1}\) is obtained by the following aggregation operation:

\[
E'_{qk1} = \text{MaxPool}(E_{i1}, E_{i2}, \ldots, E_{iM}), \quad E'_{qk1}^t \in R^{Hs}.
\]  

For the output of aggregated encoding, we add an attention mechanism to further encode the location information. The attention weight is calculation as follows:

\[
\alpha_{qki} = \frac{\exp \left( E'_{qki} \cdot u_{qk} \right)}{\sum_i \exp \left( E'_{qki} \cdot u_{qk} \right)},
\]  

where \(u_{qk}\) is calculated by

\[
u_{qk} = W_v \cdot \text{MaxPool}(E_{qk}) + b_v,
\]  

where \(W_v \in R^{HR\times HR}\), and \(b_v \in R^{HR}\). Then, we use the following attentive aggregation operation to get the document-level representation:

\[
d_{qk} = \sum_i \alpha_{qki} E'_{qki}.
\]  

Finally, we use a softmax function on \(d_{qk}\) to predict the relationship between two legal documents.

4. Experiments

4.1. Datasets and Evaluation Metrics. In this study, we use two legal text datasets, one is legal judgment document crawled from "China Judgements Online" and our topical encoding module is trained on this dataset. The other one is the LeCARD open-source dataset provided by Tsinghua University, and our BERT-LF model is experimented on the dataset of LeCARD.

The crawled legal judgment documents contain about 3.6 million legal judgment documents, covering more than 100 kinds of charges, of which the charge distribution with more than 3500 is shown in Table 1.

LeCARD is a legal case retrieval dataset in China’s legal system. It consists of 107 query cases and 10700 candidate cases, which are selected from more than 43000 criminal judgment corpora in China. The dataset is based on a series of key factors combined with subjective and objective evaluation as the correlation judgment standard. In order to ensure the diversity of cases, the dataset adopts sampling strategy, containing common query cases and controversial query cases.

4.2. Baseline Methods and Experimental Settings. We compared our model with the following baseline models:

4.2.1. Traditional Bag-of-Words Retrieval Models. We chose the traditional bag-of-words retrieval models including BM25, TF-IDF, and LIMIR, following the previous work [15]. And all parameters of these three models are set to default values in an existing package [43].

4.2.2. Neural Network Model. We compared our model with BERT-PLI [9] since it is BERT-based model and solved the problem of long text of legal cases; most importantly, our model is an improvement based on BERT-PLI. For the baseline module of BERT-PLI, we set \(N=2\) and \(M=8\), \(HB=768\).
Table 1: Distribution of partial charges for legal judgment document data.

| Charge name                        | Number   | Charge name                               | Number   |
|------------------------------------|----------|------------------------------------------|----------|
| Larceny                            | 798577   | Intentionally destroying possessions      | 24963    |
| Dangerous driving crime            | 625856   | Bribery                                   | 21368    |
| Intentional injury crime           | 394277   | Illegal business crime                    | 21047    |
| Traffic accident crime             | 292823   | Contract fraud crime                      | 20101    |
| Fraud                              | 121544   | Extortion crime                           | 19560    |
| Providing venues for drug users    | 110178   | Affray crime                              | 18565    |
| Defiance and affray crime          | 106057   | Corruption crime                          | 18199    |
| Robbery                            | 59913    | Official embezzlement crime               | 16772    |
| Casino crime                       | 59039    | Negligence causing death crime            | 11193    |
| Disrupting public service crime    | 44303    | Rape crime                                | 10447    |
| Credit card fraud crime            | 38564    | Offering bribes crime                     | 8841     |
| Illegal detention crime            | 35574    | Crime of refusing to execute judgments or orders | 5588 |
| Illegally holding drugs crime      | 33392    | Negligently causing serious accident crime | 5530    |
| Deforestation crime                | 34297    | Crime of unlawful intrusion into residence | 3660    |
| Gambling crime                     | 25304    | Nongovernmental staff bribery crime       | 3581     |

As for RNN, HR is set as 256 and only one hidden layer is used. During the training process, we use the Adam optimizer and set the start learning rate as 2e-5.

In our BERT-LF model, the parameter settings are as follows: for legal feature encoding module, we set the total number of paragraphs for query document \( N = 2 \) and the total number of paragraphs for candidate document \( M = 8 \), \( HB = 768 \), which is determined by the size of the BERT hidden vector. As for GRU, HR is set as 256 and only one hidden layer is used. In training set, 10% queries from the training set and all of their candidates are treated as the validation set. We train the model on the training data left for training set and all of their candidates are treated as the validation set. We train the model on the training data left for 40 epochs and select the best model in the training process according to the precision measure on the validation set. During the training process, we use the Adam optimizer and set the start learning rate as 2e-5. For LDA model in topical encoding module, we set the quantity of topic \( v = 7 \).

5. Results and Discussion

For evaluation, we use two metrics including precision and ranking following literature [15]. Precision metrics include P@5, P@10, and mean average precision (MAP), and ranking metrics include NDCG@10, NDCG@20, and NDCG@30.

5.1. Overall Results. Comparison results between models are shown in Table 2. The comparison among the traditional three bag-of-words retrieval models show that LMIR performs best among the precision metrics, including P@5, P@10, and MAP, which is consistent with the conclusion of literature [15]. Under the same experimental conditions of this study, LMIR performed best in the three bag-of-words models. However, traditional retrieval models are difficult to handle long text retrieval, and the input length of these models is limited and cannot represent documents well. BERT-PLI outperforms traditional bag-of-words retrieval models in all ranking metrics, and it is structurally able to consider the entire case document and has better semantic understanding than traditional models. And BERT-LF is the best in all six indicators, and it not only considers the completed case documents and improves the semantic understanding ability but also adds the topic model and entity model to logically judge and analyze the legal elements between paragraphs.

5.2. Model Ablation. In order to further analyze the effects of each module, we conducted ablation experiments, removing the gain embedding and gain mask from BERT-LF, both or one at a time, and observe the impact on the performance compared to the full model. The experimental results are shown in Table 3. Only the topical encoding module is represented by SEM-TP. Only the legal entity encoding module is represented by SEM-EE. Only the semantic encoding module is represented by SEM-EC, the topical encoding added by the legal entity encoding module is represented by SEM-TE, the topical encoding added by the semantic encoding module is represented by SEM-T, and the legal entity encoding added by the semantic encoding module is represented by SEM-E. First, the ablations of the main components result in performance declining, verifying the effectiveness of these components for BERT-LF. SEM-EE and SEM-TE achieve a very large drop, indicating that using only legal element entities or paragraph topics of paragraphs cannot represent the entire text, and the semantic module is very important in the function of text encoding in this experiment. In addition, the performance of SEM-EC (LSTM) and SEM-EC (GRU) also drops significantly, which proves that the addition of topic model and legal element entity model improves the accuracy of paragraph logic judgment. Second, SEM-T (LSTM) and SEM-T (GRU) only use semantic encoding and topic encoding, with reduced accuracy. Third, the performance of SEM-E (LSTM) and SEM-E (GRU) drops slightly, which indicates that the encoding
of text topics cannot be ignored either. In the sequence encoding part of semantic module, LSTM and GRU are verified, respectively; the result showed that GRU has better effect, which is consistent with the conclusion of literature [9].

### 6. Conclusions

In this study, we proposed a model BERT-LF for similarity case retrieval based on the legal facts; our model combined the topic distribution and legal entity facts to make the document representation vector more suitable for legal scenarios. The study adopts the architecture of cutting and aggregation on paragraph, divides the long legal text into short paragraphs according to the logical order of the case, and then represents the query-candidate paragraph pairs through the BERT-based text encoding method. On the one hand, we can use the powerful semantic encoding ability of BERT; on the other hand, we can solve the problem of long text coding of legal cases. In order to accurately excavate the legal elements in legal cases, this study excavates several legal entities that have a decisive impact on the case judgment, including charges, crime, types of criminal entity, criminal consequences, compensation behavior, and reconciliation. Through convolution neural network and attention mechanism, it not only encodes the position information of legal semantics in paragraphs but also logically judges and strengthens the legal elements between paragraphs. The experimental results demonstrate that our approach is effective in legal case retrieval and the combination with topic distribution and legal entity facts can further improve models for this task.

### Data Availability

The dataset used to support the topic encoding module of this study is available from the corresponding author upon request.

### Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

### Authors' Contributions

Weifeng Hu and Siwen Zhao contributed equally to this work.

### Acknowledgments

This work was supported by the National Key R&D Program of China under Grant 2020YFC0833303 and the Major Project of Independent Innovation in Qingdao 21-1-2-18-xx.

### References

[1] T. Bench-Capon, M. Araszkiewicz, K. Ashley et al., “A history of AI and law in 50 papers: 25 years of the international conference on AI and law,” *Artificial Intelligence and Law*, vol. 20, no. 3, pp. 215–319, 2012.

[2] M. V. Opijnen and C. Santos, “On the concept of relevance in legal information retrieval,” *Artificial Intelligence and Law*, vol. 25, no. 1, pp. 65–87, 2017.

[3] Z. Cai, Z. He, X. Guan, and Y. Li, “Collective data-sanitization for preventing sensitive information inference attacks in social
networks,” *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.

[4] Z. P. Cai and X. Zheng, “A private and efficient mechanism for data uploading in smart cyber-physical systems,” *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[5] Z. P. Cai and Z. B. He, “Trading private range counting over big IoT data,” in *The 39th IEEE International Conference on Distributed Computing Systems*, pp. 144–153, Dallas, TX, USA, 2019.

[6] D. W. Oard and W. Webber, “Information retrieval for e-discovery,” *Foundations and Trends in Information Retrieval*, vol. 7, no. 2–3, pp. 99–237, 2013.

[7] P. Bhattacharya, K. Ghosh, S. Ghosh, and A. Pal, *Overview of the FIRE 2019 AILA Track: Artificial Intelligence for Legal Assistance*, FIRE (Working Notes), 2019.

[8] J. Rabelo, M.-Y. Kim, R. Goebel, M. Yoshioka, Y. Kano, and K. Satoh, “A Summary of the COLIEE 2019 Competition,” in *JSAI International Symposium on Artificial Intelligence*, pp. 34–49, Springer, Cham, 2019.

[9] Y. Shao, J. Mao, Y. Liu et al., “BERT-PLI: modeling paragraph-level interactions for legal case retrieval,” in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence Main track*, pp. 3501–3507, Yokohama, 2020.

[10] C. Xiao, X. Hu, Z. Liu, C. Tu, and M. Sun, “Lawformer: a pre-trained language model for Chinese legal long documents,” *AI Open*, vol. 2, pp. 79–84, 2021.

[11] Z. A. Yilmaz, W. Yang, H. Zhang, and J. Lin, “Cross-Domain Modeling of Sentence-Level Evidence for Document Retrieval,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 3490–3496, Hong Kong, China, 2019.

[12] Z. Dai and J. Callan, “Deeper text understanding for ir with contextual neural language modeling,” in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 985–988, Paris, France, 2019.

[13] Z. Wu, J. Mao, Y. Liu et al., “Leveraging passage-level cumulative gain for document ranking,” in *Proceedings of The Web Conference 2020*, pp. 2421–2431, Taipei, Taiwan, 2020.

[14] C. Li, A. Yates, S. MacAvaney, B. He, and Y. Sun, “PARADE: passage representation aggregation for document ranking,” 2020, https://arxiv.org/abs/2008.09093.

[15] Y. Ma, Y. Shao, Y. Wu et al., “LeCaRD: a legal case retrieval dataset for Chinese law system,” in *SIGIR ’21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021.

[16] X. Ben, Y. Ren, J. Zhang et al., “Video-based facial micro-expression analysis: a survey of datasets, features and algorithms,” *Institute of Electrical and Electronics Engineers transactions on pattern analysis and machine intelligence*, vol. PP, p. 1, 2021.

[17] X. Ben, P. Zhang, Z. Lai, R. Yan, X. Zhai, and W. Meng, “A general tensor representation framework for cross-view gait recognition,” *Pattern Recognition*, vol. 90, pp. 87–98, 2019.

[18] X. Ben, C. Gong, P. Zhang, X. Jia, Q. Wu, and W. Meng, “Coupled patch alignment for matching cross-view gait,” *IEEE Transactions on Image Processing*, vol. 28, no. 6, pp. 3142–3157, 2019.

[19] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[20] X. Ben, C. Gong, P. Zhang, R. Yan, Q. Wu, and W. Meng, “Coupled bilinear discriminant projection for cross-view gait recognition,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 3, pp. 734–747, 2020.

[21] X. Yang, F. Feng, W. Ji, M. Wang, and T. S. Chua, “Deconfounded video moment retrieval with causal intervention,” in *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021.

[22] X. Yang, S. Wang, J. Dong, J. Dong, M. Wang, and T. S. Chua, “Video moment retrieval with cross-modal neural architecture search,” *IEEE Transactions on Image Processing*, vol. 31, pp. 1204–1216, 2022.

[23] X. Yang, J. Dong, Y. Cao, X. Wang, M. Wang, and T. S. Chua, “Tree-augmented cross-modal encoding for complex-query video retrieval,” in *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pp. 1339–1348, Xi’an, China, 2020.

[24] J. Dong, X. Li, C. Xu et al., “Dual encoding for video retrieval by text,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, 2021.

[25] S. E. Robertson and S. Walker, “Some simple effective approximations to the 2-poission model for probabilistic weighted retrieval,” in *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 232–241, London, 1994.

[26] G. Salton and C. Buckley, “Term-weighting approaches in automatic text retrieval,” *Information Processing & Management*, vol. 24, no. 5, pp. 513–523, 1988.

[27] J. M. Ponte and W. B. Croft, “A language modeling approach to information retrieval,” in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 275–281, Melbourne, Australia, 1998.

[28] A. Mandal, R. Chaki, S. Saha, K. Ghosh, A. Pal, and S. Ghosh, “Measuring similarity among legal court case documents,” in *Proceedings of the 10th annual ACM India compute conference*, pp. 1–9, Bhopal India, 2017.

[29] N. P. A. Vo, C. Privaunt, and F. Guillot, “Experimenting word embeddings in assisting legal review,” in *International Conference on Artificial Intelligence and Law*, pp. 189–198, London, United Kingdom, 2017.

[30] A. Mandal, K. Ghosh, A. Bhattacharya, A. Pal, and S. Ghosh, *Overview of the FIRE 2017 IRLeD Track: Information Retrieval from Legal Documents*, FIRE (Working Notes), 2017.

[31] X. Zheng and Z. P. Cai, “Privacy-preserved data sharing towards multiple parties in industrial IoTs,” *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 38, no. 5, pp. 968–979, 2020.

[32] L. Li, A. Zhu, and T. Su, “Research and implementation of an improved VSM-based text similarity algorithm,” *Computer Applications and Software*, vol. 29, no. 2, pp. 282–284, 2012.

[33] G. Huang, C. Guo, M. J. Kusner, Y. Sun, F. Sha, and K. Q. Weinberger, “Supervised word mover’s distance,” *Advances in Neural Information Processing Systems*, pp. 4862–4870, 2016, https://proceedings.neurips.cc/paper/2016/hash/10c66082c124f0a63f4886f5e516e0-Abstract.html.
[34] B. Lv and W. L. Hou, “Typical case recommendation of court texts based on topic model,” Microelectronics & Computer, vol. 35, no. 2, pp. 128–132, 2018.

[35] L. X. Xiang, Design and Implementation of Referee Document Recommendation System Based on Natural Semantic Processing, Nanjing University, China, 2015.

[36] Y. Wang, J. Ge, Y. Zhou et al., “Topic model based text similarity measure for Chinese judgment document,” in International Conference of Pioneering Computer Scientists, Engineers and Educators, pp. 42–54, Singapore, 2017.

[37] W. C. Deng, Research on Judicial Intelligence Based on Deep Learning, School of Computer Science and Technology, Harbin Institute of Technology, 2017.

[38] L. J. Li, Computing Document Similarity for the Legal Case Retrieval, School of Computer Science and Technology, Nanjing Normal University, 2018.

[39] B. Y. Liu, S. Li, L. Ye, and H. Zhang, “Similar case recommendation algorithm based on legal elements,” Intelligent Computer and Applications, vol. 11, no. 6, pp. 1–4, 2021.

[40] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “BERT: pre-training of deep bidirectional transformers for language understanding,” 2018, https://arxiv.org/abs/1810.04805.

[41] Z. P. Cai, Z. B. Xiong, H. H. Xu, P. Wang, W. Li, and Y. Pan, “Generative adversarial networks,” ACM Computing Surveys, vol. 54, no. 6, pp. 1–38, 2021.

[42] Y. Lyu, Z. Wang, Z. Ren et al., “Improving legal judgment prediction through reinforced criminal element extraction,” Information Processing & Management, vol. 59, no. 1, p. 102780, 2022.

[43] R. Rehurek and P. Sojka, Gensim-statistical semantics in python, Retrieved from genism.org, 2011.