Journal of Combinatorial Mathematics
and Combinatorial Computing
www.combinatorialpress.com/jcmcc

Graphical Learning Algorithm Based Analysis of Entity Relationships in Legal Texts

Amin Wang[1,✉]

[1] *Institute of Marxism, Zhengzhou Tourism Collegea, Zhengzhou, Henan, 451464, China*

ABSTRACT

The continuous improvement of judicial construction has led to the emergence of a large amount of judicial data on the Internet, and how to make full use of judicial data to promote judicial openness, fairness and efficiency has become an important issue in the construction of judicial informatization. In the article, the word vector generation technique is used to obtain the annotation sequence of legal text, and then the BiLSTM model is combined with the CRF model to realize the recognition of legal text entities, and the Adam algorithm is used to optimize the training of the model, so as to improve the recognition effect of the model on legal text entities. The GCN model in the graph representation learning algorithm is introduced, and the legal text entity recognition results are used as inputs for the construction of sequential and semantic relationships, and the GCN-BiLSTM model for legal text entity relationship extraction is constructed by combining the graph representation attention network and the BiLSTM model. Based on the self-constructed legal text dataset, the validation analysis of the above model is carried out through simulation experiments.The accuracy of the BILSTM-CRF model in legal text entity recognition is 85.67%, which is 7.35% higher than that of the single LSTM-CRF model. The GCN-BiLSTM model improves its accuracy by 2.14 percentage points compared with the CasRel model in extracting the entity relationships of legal texts with multi-entity overlapping. Combined with the legal text entity relationship extraction results, the knowledge map of legal cases can be constructed to provide accurate knowledge relationship support for sorting out the veins of legal cases.

Keywords: word vector generation, BiLSTM model, CRF model, graph representation learning, GCN model, legal text entity

1. Introduction

Graphical learning algorithms, also known as graph representation learning algorithms, is a method of representing data as graph structures and learning them, which is widely used in the fields of data mining, machine learning, and social networks. It can effectively deal with unstructured data and

✉ Corresponding author.
E-mail address: ldlj2024@126.com (A. Wang).
Received 20 December 2024; Revised 05 February 2025; Accepted 25 March 2025; Published Online 16 April 2025.
DOI: 10.61091/jcmcc127b-063

reveal the features and patterns of data from global and local perspectives. It is of great significance in the analysis of entity relationship of legal text [1-2].

With the development of the Internet, there are more and more legal texts, how to quickly extract key information from massive legal texts to help people understand the legal knowledge and improve the efficiency of case handling has become a current research hotspot [3-5]. Entities and entity relationships are the key information in the text, and extracting the entities and entity relationships in the legal case text can provide data support for constructing case knowledge graph [6-8]. Currently, the main problems of entity-relationship extraction applied in the legal field are that there is no clear boundary character to identify the word boundary in Chinese case text, and lexical information usually plays a crucial role in determining the entity boundary [9-12]. The current entity-relationship extraction model lacks the access to the word feature information in the sentence, and the structure is complicated. In addition, some of the case texts have overlapping entity relations, i.e., one entity corresponds to multiple relations, which affects the accuracy of entity relation extraction [13-16].

With the acceleration of the process of ruling the country according to the law, various laws and regulations have been introduced one after another, but there are numerous cases of illegal and criminal behavior every year, which leads to the intensification of the contradiction of "too many cases, too few people". Starting from the annotation of legal text, the article uses the Skip-gram model to obtain the word vector sequence of the legal text, and obtains the contextual information of the legal text through the BiLSTM model, and then combines the CRF model to globally optimize the labels, so as to improve the entity recognition effect of the legal text. Based on the GCN model in the graph representation learning algorithm, the entity recognition results are used as inputs to construct sequence relations and semantic relations, and then the graph representation attention network is introduced to improve the feature extraction ability, and then the BiLSTM model is combined to obtain the contextual features of the entity relations of the legal text, and then the legal text entity relation extraction model based on GCN-BiLSTM is constructed. Based on the self-constructed legal text dataset, the effectiveness of the above model is explored, and the legal text knowledge graph is constructed by combining the results of legal text entity extraction to illustrate the feasibility and research value of the method.

2. Methods of identifying legal text entities

With the accelerating pace of the construction of the rule of law and the rapid development of artificial intelligence technology, the application of artificial intelligence technology in the judicial field is becoming more and more mature. Judicial documents are different from ordinary texts, which usually contain a large number of legal terminology and have multiple meanings, etc. Intelligent labeling and extraction of the relevant entities of the transcripts and other textual materials can significantly improve the efficiency of the case, and the relevant research is of great significance. However, the traditional Chinese named entity recognition model has the problem of insufficient extraction ability, in this regard, the development of deep learning-based legal text entity recognition helps to better realize the extraction and analysis of legal text entity relationship.

2.1. Legal Text Annotation and Word Vector Generation

2.1.1 Marking of legal texts. In the actual legal provisions, all legal provisions are set up around the establishment of a legal relationship to begin with, the establishment of a legal relationship, first of all, need to follow certain principles, such as the establishment of a relationship involving property interests, need to follow the principle of fairness. The construction of a legal relationship requires three elements, namely, the subject of the legal relationship, the object of the legal relationship and the content of the legal relationship. The body of knowledge in the legal field is shown in Figure 1. In different legal relationships, the subject and object are referred to differently. The core part of the

content of the legal relationship contains the rights and obligations of the subject and object in the legal relationship.

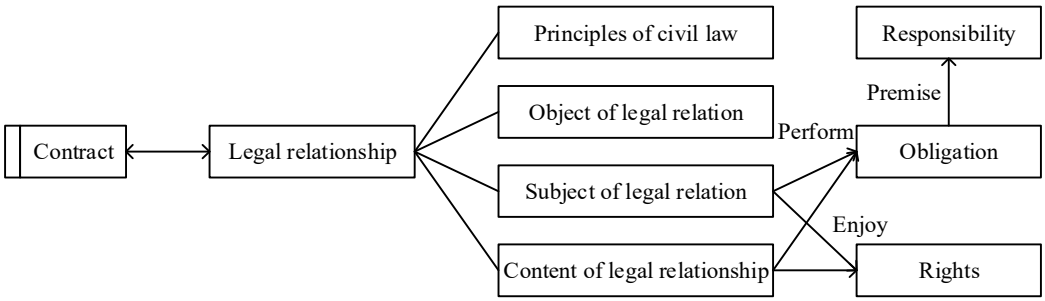

Fig. 1. Legal domain knowledge system

In this paper, the "BIO" annotation strategy is adopted, combined with the entity type right (right) in the legal field, which is denoted as "RIG", and the annotated characters can be obtained from the annotated characters as the right entity in the input text, and the obtained entity identification tags are set as role (RUL), contract (COT), moral principle (PRI), data document proof (MAT), legal relationship (CON), right (RIG), and responsibility (DUT) , Obligation (OBL), Legal Acts (BEH), Money (MON), Definitions (DEF), Illegal and Criminal Acts (ILL).

Entity labeling was performed on the original legal text using the annotation tool Wizard Labeling Assistant to "." , ";", ":" for each legal text clause to ensure the sample size for subsequent entry into model training. After the legal text is divided into sentences according to the defined symbols, there are 1578 legal statements in total, and the legal statements are labeled with entities, and finally the entity labeled ann.file of the law is obtained. Combining the positional information of term entities and the original legal txt.text, we get the named entity recognition dataset of legal text labeled with "BIO" annotation strategy (Law).

2.1.2 Word Vector Generation Techniques. In this paper, we propose a multi-feature word vector generation model is constructed based on Skip-gram algorithm, and use the negative sampling algorithm to assist the training of the model, the main principle is to use the center word to predict the background words in the upper and lower range of the text [17]. Skip-gram model is a neural network structure consisting of input, mapping, and output layers, and the training of the model is essentially the training of the neural network. That is, the weight parameters of the neurons are constantly adjusted by the gap between the input sample data and the real value, so as to improve the accuracy of the model.

Assume that the size of the dictionary is N and the set of indexes of the dictionary is $V = \{0,1,\ldots,|V|-1\}$. Given a corpus C, the words in corpus C are mapped one-to-one with the words in the dictionary and a word index is created. Let the word indexed as i in the dictionary be w_i. This word has two vector representations, the background word $u_i \in \square^d$ and the center word $v_i \in \square^d$, where d denotes the vector dimension. Based on this theory, let the word w_c with index c in the dictionary be the center word and let the word w_o with index o in the dictionary be the background word, then the conditional probability of generating the background word w_o given the center word w_c is:

$$P(w_o \mid w_c) = \frac{\exp(u_o^{\bullet} v_c)}{\sum_{i \in V} \exp(u_i^{\bullet} v_c)} \tag{1}$$

The loss function of the model is:

$$-\sum_{t=1}^{T}\sum_{-m\le j\le m, j\neq 0}\log P(w^{(t+j)}\mid w^{(t)}) \tag{2}$$

where T denotes the length of the text sequence, w^{θ} is the word corresponding to the moment of time step t, and m is the size of the background window.

In the process of minimizing the loss function, the Stochastic Gradient Descent (SGD) algorithm is used to calculate the minimum value, and the process of training the model using this method is as follows: firstly, a subsequence is randomly selected in each iteration and the loss value of the sample is calculated, and then the gradient value is calculated using this sample, and finally, the parameters of the model are updated according to the results obtained. In this case, the key to calculating the gradient value of the loss function is to find the logarithmic conditional probability of the background word to the center word, then:

$$\log P(w_o \mid w_c) = u_o^{\cdot} v_c - \log(\sum_{i\in V}\exp(u_i^{\cdot} v_c)g) \tag{3}$$

Differentiating the above equation yields the gradient of any center word vector v_e, computed as follows:

$$\begin{aligned}\frac{\partial \log P(w_o \mid w_c)}{\partial v_c} &= u_o - \frac{\sum_{j\in V}\exp(u_j^{\cdot} v_c)u_j}{\sum_{i\in V}\exp(u_i^{\cdot} v_c)}\\ &= u_o - \sum_{j\in V}\left(\frac{\exp(u_j^{\cdot} v_c)}{\sum_{i\in V}\exp(u_i^{\cdot} v_c)}\right)u_j\\ &= u_o - \sum_{j\in V}P(w_j \mid w_c)u_j\end{aligned} \tag{4}$$

Similarly, the gradient of any background word vector u_c can be derived in this way.
In the process of training the model, because the above formula will introduce the background word vector of the whole dictionary in the process of calculation, and at the same time accumulate the terms of the dictionary size into the loss function. This makes the computational resource overhead become large, and greatly extends the training time of the model, seriously affecting the efficiency of model training. To address this situation, this model uses the negative sampling method to approximate the model training.

2.2. BiLSTM and CRF modeling

2.2.1 BiLSTM modeling. Long Short-Term Memory (LSTM) networks are a class of temporal recurrent neural networks obtained by structural modification of Recurrent Neural Networks (RNN), which effectively solves the problems of gradient explosion and gradient vanishing that often occur in RNN, and is able to better capture long dependencies in the front and back texts of legal texts.

The single base structure of LSTM internally includes three gate structures and one unit state. The gate structures are an input gate, a forgetting gate, and an output gate, which control the transfer of information, allowing the LSTM network to autonomously decide when to read in, discard, or output information to better process long text sequences [18].

The input gate determines the effect of new input information on the state of the cell at the current state of time and is calculated as:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{5}$$

where σ is the Sigmoid activation function, W_i, b_i is the weight matrix and bias vector of the input

gate, respectively, h_{t-1} is the hidden state of the $t-1$ time step, and x_t denotes the input sequence of the current time step.

The forgetting gate determines how much of the previous time unit state is retained in the current time state and determines how much the past time unit state affects the current one, and is calculated as:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{6}$$

where W_f and b_f are the weight matrix and bias vector of the forgetting gate, respectively.

The output gate determines which information about the cell cell state will be output and determines how much the current cell state affects the final hidden layer, and is calculated as:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{7}$$

where w_o and b_o are the weight matrix and bias vector of the output gate, respectively.

The cell state stores the long-term state information of the text and is calculated as:

$$C_t = f_t * C_{t-1} + i_t * tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{8}$$

The hidden layer state update is represented as:

$$h_t = o_t * \tanh(C_t) \tag{9}$$

where $*$ denotes the element-by-element multiplication, $tanh$ is the hyperbolic tangent activation function, and w_c and b_c are the weight matrices and bias vectors of the state units.

Bidirectional Long Short-Term Memory Network (BiLSTM) is further improved on the basis of LSTM by considering both the information of the current time step and the subsequent time step of the input sequence.BiLSTM consists of two independent LSTM networks, which process the input sequence according to the forward and inverse sequences, respectively, and obtains the forward LSTM $\vec{h}_t$ and the inverse one $\overleftarrow{h}_t$ denoted as:

$$\vec{h}_t = \mathrm{LSTM}(x_t, \vec{h}_{t-1}) \tag{10}$$

$$\overleftarrow{h}_t = \mathrm{LSTM}(x_t, \overleftarrow{h}_{t+1}) \tag{11}$$

The final hidden layer representation is obtained by concatenating the two bidirectional LSTM outputs h_t. Then:

$$h_t = \left[\vec{h}_t, \overleftarrow{h}_t\right] \tag{12}$$

Thus, the model can be made to obtain the information of the sequence's preamble and subsequent states at each time point. Thus BiLSTM is able to understand the contextual information more comprehensively and is more suitable for tasks such as predicate recognition, semantic role labeling, etc. than LSTM.

2.2.2 CRF model. Conditional Random Fields (CRF) is a discriminative undirected graph model with a conditional distribution of associative graphical structure $P(y \mid x)$ that does not require direct modeling of the dependencies of input variables x [19].

Let X and Y be random variables, and the Markov random field of output Y conditional on input variable X is called the conditional random field, X is an observation of Y. Let $G = (V, E)$ be an undirected graph, and the nodes of the undirected graph correspond one-to-one with the elements in the random variable Y, assuming that each variable in Fig. G satisfies Markovianity, i.e.:

$$P(Y_v \mid X, Y_{V\setminus\{v\}}) = P(Y_v \mid X, Y_{n(v)}) \tag{13}$$

where Y_v denotes the random variable corresponding to node v, and $n(v)$ denotes the node that is

edge-connected to node v.

Since the CRF is essentially a Markov random field given a sequence of observations X, the conditional probability of the conditional random field can be written as:

$$P(Y \mid X) = \frac{1}{Z(X)} \prod_{c} \Psi_c(X, Y_c) \tag{14}$$

where $Z(X) = \Sigma_Y \prod_C \Psi_c(X, Y_c)$ At the same time, influenced by the maximum entropy model, the potential function is generally defined in the form of an exponential and an eigenfunction is introduced:

$$\Psi_c(X, Y_c) = \exp\left(\sum_k \theta_k f_k \left(C, Y_c, X \right) \right) \tag{15}$$

where θ_k is the weight corresponding to the characteristic function and f_k denotes a Boolean characteristic function.

Therefore, the conditional probability of a conditional random field has the general form:

$$P(Y \mid X) = \frac{1}{Z(X)} \exp\left(\sum_k \theta_k f_k \left(C, Y_c, X \right) \right) \tag{16}$$

$Z(X) = \sum_Y exp(\sum_k \theta_k f_k(C, Y_c, X))$ of them.

2.3. BiLSTM-CRF modeling

2.3.1 Model framework and design. In this paper, the BiLSTM-CRF model is used for the task of sequence annotation and named entity recognition of legal texts, and the structure of the BiLSTM-CRF model is shown in Figure 2. The probability of each word corresponding to each label is obtained using the BiLSTM model using SoftMax for normalization, however, each label does not exist independently, and there are certain constraints between them. While CRF can better learn the dependency relationship between each label, global optimization can make the annotation processing more accurate and efficient.

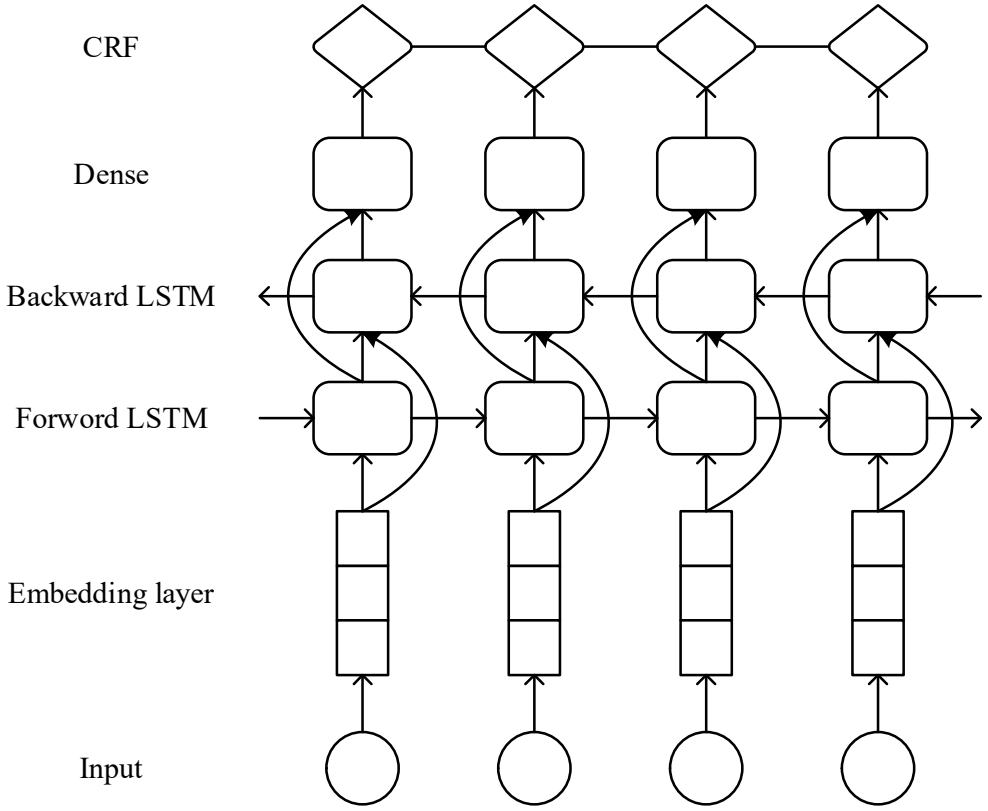

Fig. 2. The structure of the BiLSTM-CRF model

BiLSTM is a combination of a forward LSTM and a backward LSTM to obtain a forward t-moment hidden layer output $\vec{h}_t$ and a backward t-moment hidden layer output $\overleftarrow{h}_t$ spliced together $[\vec{h}_t, \overleftarrow{h}_t]$ to fully utilize the contextual information.

2.3.2 Optimized training of recognition models. In order to avoid the local minima problem in the model, and to improve the prediction speed and accuracy of the model, the algorithm of the model is optimized.Adam's algorithm is an extension of stochastic gradient descent method, based on the adaptive estimation of low-order matrices.The algorithm calculates the first-order moment estimation and the second-order moment estimation of the model, and according to the results, it sets different dynamic learning rates for each model, and continuously updates the model parameters through each iteration. update the model parameters.Adam not only saves the historical gradient squared exponential decay averages, but also the past gradient exponential decay averages, so the update preserves the previous direction to a certain extent.Adam's model training is optimized as follows:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \tag{17}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \tag{18}$$

where m_t is the first-order momentum of the current gradient, v_t is the second-order momentum of the current gradient, g_t is the current gradient value, β_1 is the first-order matrix exponential decay rate, and β_2 is the second-order matrix exponential decay rate. Since m_t and v_t are both zero vectors initially, the decay rate may be biased toward the zero vector at the initial time, so a bias correction is needed, and the formula is as follows:

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \tag{19}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t} \tag{20}$$

where $\hat{m}_t$ is the corrected gradient band-weighted average and $\hat{v}_t$ is the corrected gradient band-weighted biased variance.

$$\theta_{t+1} = \theta_t - \frac{\eta}{\varepsilon + \sqrt{\hat{v}_t}} \hat{m}_t \tag{21}$$

Where ε is the default value of 10^{-8} and η is the learning rate with an initial value of 0.01.

3. Extraction of legal text entity relationships in conjunction with graphical representations of learning

At present, courts at all levels have accumulated large-scale electronic case files and other data resources in the process of informationization construction, which contain massive case information, judicial knowledge and judges' experience in handling cases, and are important court big data assets. How to accurately and effectively extract the legal elements needed by judges or case handlers in the massive judicial data is the first condition of judicial informatization construction. Legal elements refer to key information such as plaintiff, defendant, and case in legal documents, and to extract these key information from the documents, it mainly relies on named entity recognition and entity relationship extraction techniques in the field of natural language processing. Recognizing and extracting legal entity relationships in legal documents in the judicial field and efficiently and accurately transforming legal documents in text form into structured information is one of the urgent needs for the construction of intelligent courts.

3.1. Graph Representation Learning and GCN Modeling

3.1.1. Graph representation learning algorithm. Graph representation learning is a technique for studying how to learn effective representations from graph-structured data. Graph-structured data is a common data type that can be used to represent relationships (edges) between entities (vertices). However, traditional machine learning methods often cannot directly deal with graph-structured data, and graph-structured data has complex topologies and relationships, which makes learning effective graph representations challenging [20].

Graph representation learning can be formally defined as follows:

Given a graph $G = (V, E)$, where V is a collection of nodes and $E \subseteq V \times V$ is a collection of edges. Each node $v \in V$ may have a corresponding attribute feature $x_v \in \mathbb{R}^d$. The goal of graph representation learning is to learn a mapping function $f : V \to \mathbb{R}^k$ that projects a node v to a low-dimensional vector $z_v \in \mathbb{R}^k$ while preserving the graph structure and attribute information.

Graph representation learning can be categorized as follows:

1) Factorization-based approach. The graph representation learning problem is transformed into a low-rank matrix decomposition problem, where the node representations are obtained by decomposing the adjacency matrix or Laplace matrix of the graph, and the node representations are learned by optimizing the objective function.

2) Random-walk based approach. Generate node sequences by simulating the random wandering process on the graph, and then learn the node representations using sequence modeling techniques. Classical algorithms include DeepWalk and Node2Vec.

3) Neural network-based methods. Learning the representation of nodes in a graph by designing a specific neural network structure, including methods based on graph convolutional neural networks (GCN).

In order to achieve effective extraction of legal text entity relationship, this paper mainly adopts

graph convolutional neural network in graph representation learning, combines it with BiLSTM model to achieve legal text entity relationship extraction, and provides a new reference for optimizing legal text data and promoting judicial intelligence.

3.1.2 Graph Convolutional Neural Networks. The graph neural network (GCN) model equates information aggregation operations on graph data to signal filtering on graph data, and this class of methods transforms the graph data to the graph frequency domain via the graph Laplace matrix, which is then processed using the corresponding convolution kernel, i.e., a matrix filter [21]. Namely:

$$(f * g)_G = U((U^T g) \odot (U^T f)) \tag{22}$$

Where g is the graph convolution kernel function, which is the filter in the spectral domain, f is the initial eigenvectors of the nodes, and U is the matrix consisting of the eigenvectors of the graph Laplace matrix. The spectral domain method first uses the graph Fourier transform to map g and f to the frequency domain for Hadamard product, and after obtaining the convolution result equivalent to the feature aggregation operation in the frequency domain, it is finally mapped to the original spatial domain using the graph inverse Fourier transform. Moreover, another representation of the above equation is given as:

$$(f * g)_G = U\left[diag(\hat{g}(\lambda_l))\right]U^T f, l \in [1, n] \tag{23}$$

where $diag(\hat{g}(\lambda_l))$ is the diagonal matrix with $\hat{g}(\lambda_l)$ as an element and $\left[diag(\hat{g}(\lambda_l))\right]$ is the convolution kernel. λ_l is the lth eigenvalue of the graph Laplace matrix.

After a series of simplifications, the form of the graph neural network spectral method evolves as:

$$y = \sigma(U g_\theta(\Lambda) U^T x) \tag{24}$$

where $\Lambda = diag(\lambda_l)$, $g_\theta(\Lambda)$ are convolution kernels while $\sigma(\cdot)$ is a nonlinear activation function and x is a node feature. Then:

$$g_\theta(\Lambda) = \sum_{k=0}^{K-1} \beta_k T_k(\tilde{\Lambda}) \tag{25}$$

where β_k represents the coefficients of the Chebyshev polynomials of the k nd order, $T_k(\cdot)$ is the Chebyshev polynomials of the k th order, while $\tilde{\Lambda} = \frac{2}{\lambda_{max}}\Lambda - I$ is the scaled diagonal matrix consisting of diagonal elements from the eigenvalues, and λ_{max} is the maximum eigenvalue of the matrix. Then:

$$y = \sigma(U\sum_{k=0}^{K-1} \beta_k T_k(\tilde{\Lambda}) U^T x) = \sigma(\sum_{k=0}^{K-1} \beta_k T_k(U\tilde{\Lambda}U^T)x) \tag{26}$$

The Laplace matrix, $L = U\Lambda U^T$, can be obtained by bringing in:

$$y = \sigma(\sum_{k=0}^{K-1} \beta_k T_k(\tilde{L})x) \tag{27}$$

where $\tilde{L} = 2L/\lambda_{max} - I$. It can be seen from Eq. (26) that the use of Chebyshev polynomials to form the convolution kernel eliminates the need for further eigenvector decomposition of the Laplace matrix, reducing the computational complexity of the spectral method. On this basis, computing the Chebyshev polynomials using recursive Eq. $T_k(\tilde{L}) = 2\tilde{L}T_{k-1}(\tilde{L}) - T_{k-2}(\tilde{L})$ can further reduce the computational complexity by initializing $T_0(\tilde{L}) = I, T_1(\tilde{L}) = \tilde{L}$. The graph convolution formula for the GCN model can be expressed as:

$$H = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}X\Theta \tag{28}$$

where X is the input feature matrix, $\tilde{A} = A + I, \tilde{D} = \sum_j \tilde{A}_{ij}$, A are the adjacency matrices of the graph. After the second-order approximation operation the graph convolution neural network model has the local aggregation interpretability, the above equation can interpret the graph convolution operation as the feature aggregation operation of neighboring nodes.

3.2. Legal Text Entity Relationship Extraction Model

3.2.1 Sequence relations and semantic relations. The description of the case in the legal text involves the specific plot information of the case, and the description of the case not only contains key information such as the time, place, subject, specific events, related disputes and reasons, but also includes some descriptive non-key information. Therefore, in this paper, the graph representation learning algorithm is used for the construction of sequence relationship and semantic relationship, which lays the foundation for the extraction of entity relationship in legal text.

The Sequence Relationship Graph (SEG1) is denoted as $G_q = (V, E_q, A_q)$. Where V, E_q and A_q denote the set of nodes, the set of edges and the adjacency matrix respectively. There are two kinds of nodes in the sequence relation graph, the word nodes are the words appearing in the text $Fact = \{h_1, h_2, ..., h_i, ..., h_l\}$, and the main nodes are the representation nodes of the whole document. The edges in the text sequence relation graph are classified into two kinds: first, the edges between word nodes are generated according to a fixed-size sliding window, which connects the center node of each sliding window to other nodes; second, the edges between the document node (master node) and other word nodes. Specifically, the text sequence relationship graph is defined as:

$$V = \begin{cases} v_{n,i} \mid i \in [1,l], n \text{ is word} \\ v_{n,i} \mid i = 0, n \text{ is document} \end{cases} \tag{29}$$

$$E_q = \begin{cases} e_{ij}^{w,w} \mid i \in [1,l]; j \in [i-p, i+p] \\ e_{i,j}^{w,d} e_{i,j} \mid, i \in [1,l]; j = 0 \end{cases}, w \text{ is word}, d \text{ is document} \tag{30}$$

Where the weights of the edges $e_{ij}^{w,w}$ between the word nodes in the adjacency matrix are calculated based on the frequency and distance of occurrence of word pairs $\langle v_{n,i}, v_{n,j} \rangle$ during sliding in a fixed-size sliding window, and the weights of the edges between the word nodes and the document nodes $e_{i,j}^{w,d}$ are 1, and the weight value of the edges between each word and itself $e_{ij}^{w,w} (i = j)$ is 1.

The semantic relation graph (SEG2) is denoted as $G_m = (V, E_m, A_m)$, where V, E_m and A_m denote the set of nodes, the set of edges and the adjacency matrix, respectively. There are two kinds of nodes in the semantic relation graph, the word nodes are the words appearing in the text $Fact = \{h_1, h_2, ..., h_i, ..., h_l\}$, and the main nodes are the representation nodes of the whole document. The edges in the semantic relation graph are categorized into two kinds: one is based on the edges between words, which are determined based on the similarity of word embedded representations, and the other is the edges between the document nodes (master nodes) and other word nodes. Specifically, the semantic relationship graph is defined as:

$$V = \begin{cases} v_{n,i} \mid i \in [1,l], n \text{ is word} \\ v_{n,i} \mid i = 0, n \text{ is document} \end{cases} \tag{31}$$

$$E_m = \begin{cases} e_{i,j}^{w,w} \mid i \in [1,l]; j \in [i-p, i+p] \\ e_{i,j}^{w,d} e_{i,j} \mid, i \in [1,l]; j = 0 \end{cases}, w \text{ is word}, d \text{ is document} \tag{32}$$

Where the weight of edge $e_{i,j}^{w,w}$ between word nodes in the adjacency matrix is calculated based on word similarity, i.e., the value of cosine similarity between individual word pairs, and the weight $e_{ij}^{w,d}$ of the edge between a word node and a document node is 1, and the weight value of the edge between

each word and itself $e_{i,j}^{w,w}(i=j)$ is 1.

3.2.2 Graph Representation of Attention Networks. The GCN-BiLSTM model constructed in this study is in the form of directed graphs, but the traditional GCN faces a performance bottleneck due to the limitation of fixed weights when dealing with directed graphs. In this paper, based on sequential and semantic relations, the attention mechanism is integrated into the information transfer between different relations, which makes it capable of capturing more detailed relationships between relations and dynamically adjusting the attention weights between relations according to their characteristics and similarities. As a result, each relationship can focus on its important neighbor nodes in the process of information transfer and weightedly aggregate the feature vectors of the neighbor nodes. Therefore, this study designs an element graph attention network (EGAT) to incorporate the attention mechanism into the node information transfer process of the GCN-BiLSTM model and obtains a graph representation of the GCN-BiLSTM model.

Specifically, for target node v_i, the attention coefficients $e_{v_i v_j}$ are first computed one by one with its first-order neighbor nodes, then:

$$e_{v_i v_j} = a([Wh_{v_i} \,||\, Wh_{v_j}]), v_j \in N_{v_i} \tag{33}$$

where h_{v_i} and h_{v_j} denote the vector representations of node v_i and its first-order neighbor nodes currently involved in the computation, respectively, and N_{v_i} denotes the set of first-order neighbor nodes of node v_i. W is a shared parameter matrix for linear transformation of node vectors, || denotes the splicing operation, and $a(\cdot)$ is used to map the spliced high-dimensional features to a real number.

To ensure that the weight representation of the nodes is relatively balanced and reasonable in range, the attention coefficients are normalized using the $Softmax(\cdot)$ function, then:

$$a_{v_i v_j} = \frac{\exp(LeakyReLU(e_{v_i v_j}))}{\sum_{k \in N_{v_i}} \exp(LeakyReLU(e_{v_i v_k}))} \tag{34}$$

Here, $LeakyReLU(\cdot)$ denotes the activation function.

After that, the current node's representation is updated by weighted summation of all first-order neighboring nodes based on the computed attention coefficients. Then:

$$h_{v_i'} = \sigma\left(\sum_{v_j \in N_{v_i}} a_{v_i v_j} W h_{v_j}\right) \tag{35}$$

where $\sigma(\cdot)$ is an activation function and $h_{v_i}^{'}$ is an update feature of node v_i fused with neighborhood information.

On this basis, in order to enable the model to capture features at different levels, this paper further introduces multi-head attention. This is a self-attention based strategy that employs multiple attention heads working in parallel to enhance the fitting and representation ability of the model. The process is:

$$h_{v_i}^{'}(K) = \Big\|_{k=1}^{K} \sigma(\Sigma_{v_j \in N_{v_i}} a_{v_i v_j}^{k} W^{k} h_{v_j}) \tag{36}$$

where || denotes the splicing operation and K denotes the number of heads in multi-head attention. Each attention head in the multi-head attention mechanism obtains a vector representation $h_{v_i}^{'}$ and the final output is the splicing result of the K set of vectors.

Finally, in this paper, the maximum pooling $MaxPooling(\cdot)$ function is used to process all the elemental node vectors $h_{v_i}^{'}(K)$ in the GCN-BiLSTM model, and the most representative features are extracted from each node, which constitutes the final representation H_{g_i} of the GCN-BiLSTM model,

then:

$$H_{g_i} = MaxPooling(h'_{v_i}(K)) \tag{37}$$

3.2.3 GCN-BiLSTM modeling. Based on the sequential and semantic relations of legal texts, GCN and BiLSTM models are combined to construct a GCN-BiLSTM model for extracting entity relations of legal texts, and its model architecture is shown in Figure 3. The model mainly contains five modules, i.e., input module, sequence relationship and semantic relationship module, GCN module, BiLSTM module and output module.

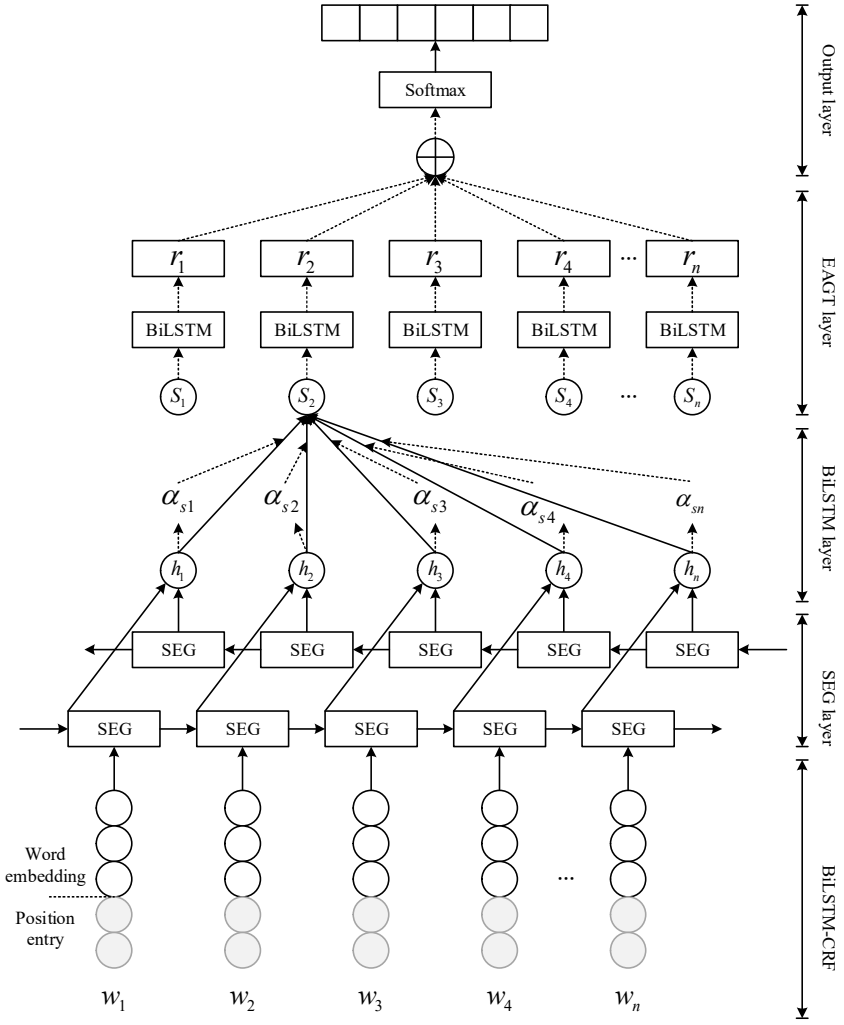

Fig. 3. The structure of the GCN-BiLSTM model

The input module is based on the legal text named entity recognition results obtained from the BiLSTM-CRF model given in the previous section, and then it enters the sequence relation and semantic relation modules for encoding, the GCN layer applies the graph convolution operation to update the node features, the BiLSTM layer aggregates the node features to get the features of the whole graph, and finally outputs the legal text entity relation extraction results through the SoftMax layer. In the training process, for each sample of legal text, their graphs each go through the same embedding module to get the final representation.

4. Legal Text Entity Relationship Identification and Extraction Validation

With the continuous development of artificial intelligence technology, the new technology has brought more possibilities for upgrading various industries. Among them, "intelligent justice" is the focus of in-depth research by major organizations. In judicial practice, large-scale judicial data, transcript information and judicial documents provide empirical guidance for studying historical cases, analyzing crimes and deciding case outcomes. How to use these data efficiently and accurately to help legal professionals in the judicial field to read and analyze the texts has become a key issue to be solved nowadays. This paper proposes a GCN-BiLSTM-based entity relationship extraction model for legal texts, which provides theoretical support for the development of intelligent justice by obtaining the relationships between entities in legal texts.

4.1. Validation of legal text entity identification

4.1.1 Substantive identification of legal texts. In order to verify the effectiveness of the BiLSTM-CRF model proposed in this paper in legal text entity recognition, the recognition validation is carried out for 12 different types of legal text entity categories given in the previous paper. For the entity recognition performance of the model, this paper adopts the common evaluation metrics of precision rate, recall rate and F1 value for evaluation in the experiments. Based on the Law dataset constructed in the previous paper, the Adam gradient descent method is utilized to train the BiLSTM-CRF model, and then the model validation is carried out on the validation set and test set. Figure 4 shows the legal text entity recognition results of BiLSTM-CRF model.

As can be seen from the figure, when the BiLSTM-CRF model established in this paper is used for legal text entity recognition, the mean values of precision rate, recall rate and F1 value on the 12 entity categories are 74.23%, 77.89% and 79.65%, respectively, and the overall recognition rate is above 70%. The legal text entity categories with the highest precision rate, recall rate and F1 value are Definition (DEF), Responsibility (DUT) and Material Instrumental Testimonial (MAT), with values of 85.58%, 83.58% and 84.98%, respectively, which are all above 80%. This indicates that the BiLSTM-CRF model designed in this paper has a high legal text entity recognition accuracy, and the legal text sequences generated by Skip-gram word vector technology, using the BiLSTM model to obtain its forward features and combining with the CRF model to globally optimize the probability of each entity labeling, and then to improve the effect of recognition of legal text entities.

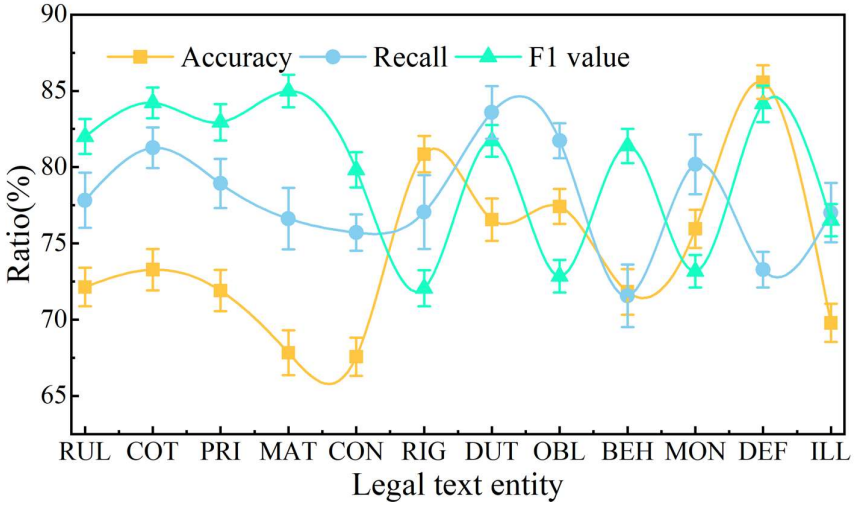

Fig. 4. Legal text entity recognition results

4.1.2 Comparison of the performance of different models. In order to validate the effect of the BiLSTM-CRF named entity model proposed in this paper, this paper adopts an experimental comparison approach, combining the current mainstream named entity recognition methods such as LSTM-CRF model, BERT-BiLSTM model, IDCNN-CRF model, and BERT-IDCNN-CRF, with the BiLSTM-CRF model proposed in this paper on the same dataset and the experimental results are evaluated by using precision, recall and F1 values. The experimental results of different models on Law dataset are shown in Table 1.

Based on the data comparison in the table, the following conclusions can be drawn:

1) In this experiment, the accuracy, recall, and F1 value of IDCNN-CRF model are 80.46%, 83.38%, and 81.49%, respectively, and those of BERT-BiLSTM model are 81.15%, 84.26%, and 82.37%, respectively, although the overall effect of BERT-BiLSTM model is better than IDCNN- CRF, but the difference between the two is not large, and the final F1 value only differs by 0.88%, but the IDCNN-CRF model takes less time to train than the BILSMT-CRF model.

2) In this experiment, the accuracy, recall, and F1 value of the LSTM-CRF model are 78.32%, 81.15%, and 80.64%, respectively, while the BERT-IDCNN-CRF model obtains an accuracy of 84.23%, a recall of 87.64%, and an F1 value of 86.15%, which is 3.08% higher than the LSTM-CRF model, 3.38% and 3.78%, respectively. It indicates that comparing with a single LSTM structure, the BERT+IDCNN model is able to achieve the extraction of contextual features of legal text from the realization of the global perspective to provide the model with labeled sequences, which effectively improves the recognition results of the model.

3) Comparing the BERT-IDCNN-CRF model, the performance of BILSTM-CRF model with word vector technique in this paper is 85.67%, 89.51%, 88.96% in accuracy, recall, and F1 value, which is 7.35%, 8.36%, and 8.32% higher than the single LSTM-CRF model, respectively. It is fully demonstrated that bidirectional LSTM can combine legal text character vectors, sentence markers and location information to encode vectors, providing word vectors incorporating more semantic information for the downstream task with greater semantic characterization capability.

In summary, the BiLSTM-CRF model designed in this paper outperforms the current mainstream named entity model in legal text entity recognition, and can obtain more accurate legal text entity recognition results, laying the foundation for legal text entity relationship extraction.

Table 1. Physical identification of different models

Model	Accuracy	Recall	F1 value
IDCNN-CRF	80.46%	83.38%	81.49%
LSTM-CRF	78.32%	81.15%	80.64%
BERT-BiLSTM	81.15%	84.26%	82.37%
BERT-IDCNN-CRF	84.23%	87.64%	86.15%
BiLSTM-CRF	85.67%	89.51%	88.96%

4.1.3 Impact of hyperparameters on the effectiveness of entity recognition. BiLSTM-CRF model in the training process is mainly used in the Adam algorithm for optimization training, the need to adjust the learning rate hyperparameters, different learning rate parameters may have a greater impact on the entity recognition effect of the model. Based on this, this paper analyzes the effect of different learning rates under the model on legal text entity recognition, entity recognition effect under different learning rates is shown in Figure 5.

When the value of learning rate is between 0 and 0.5, the precision rate, recall rate and F1 value of BiLSTM-CRF model are closer to each other, and when it is greater than 0.5, there is obviously the phenomenon of high recall and low precision rate. This is mainly because the Skip-gram word vector generation technique only focuses on the boundary information, not on the type of words, in the legal text entity recognition task. When the proportion of the word segmentation task increases, the number of recognized entities is much larger than the number of correctly recognized entities, which leads to a decrease in the precision rate. Therefore it is important to choose an appropriate learning rate for the model in this paper. By observing the F1 values, it can be found that the F1 values corresponding to different learning rate values are about 12.5 percentage points higher than the 80.64% of the LSTM-CRF model, which is because the introduction of the Skip-gram word vector generation technique can allow the model to learn additional information under the condition of ensuring that the named entity recognition task is adequately trained. -CRF model designed in this paper has certain robustness. Taking the above into account, the learning rate of the model in this paper is set to 0.2 as a way to ensure that the precision, recall and F1 values of the model have the same trend.

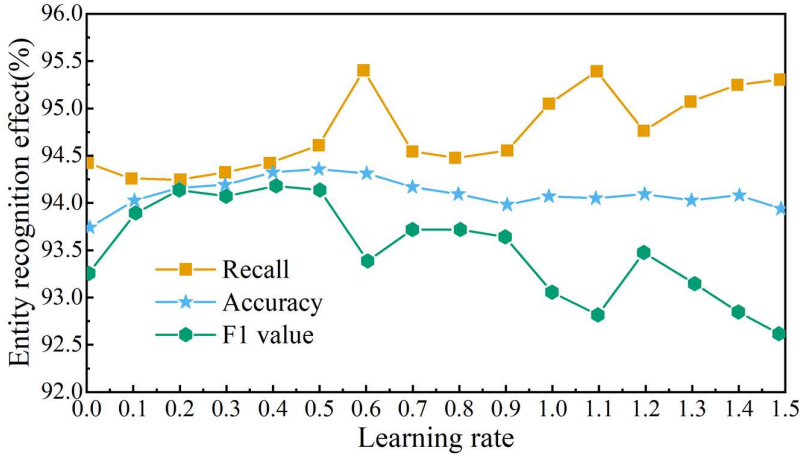

Fig. 5. Physical recognition effects under different learning rates

4.2. Extraction of legal text entity relationships

4.2.1 Comparative analysis of different models. In order to verify the effectiveness of the GCN-BiLSTM model designed in this paper in performing legal text entity relationship extraction, this paper uses the Law dataset to verify the effectiveness of each module in the model in improving the performance

of legal text entity relationship extraction. The experimental results are demonstrated using P-R curves, and their specific results are shown in Figure 6. Where SEG denotes the Sequential Relationship and Semantic Relationship module and EGAT denotes the Graph Representation Attention module.

From the figure, it can be seen that each improvement module plays a certain role in the legal text entity relationship extraction performance improvement. After adding the BiLSTM module, the precision of the improved model almost reaches the maximum among all the algorithms when the recall is about 0.021, but after that the precision decreases faster with the increase of the recall. After adding the sequence relation and semantic relation modules the precision of the improved model has some improvement compared to the original model, but the improvement is small. After further adding the graph representation attention module, the decrease of model precision is effectively alleviated with the increase of recall, and the model precision is higher than the original model and the model with some modules added after the recall is greater than 0.05. On the whole, each module in the GCN-BiLSTM model designed in this paper has certain improvement on legal text entity relationship extraction, which can obviously enhance the effect of legal text entity relationship extraction.

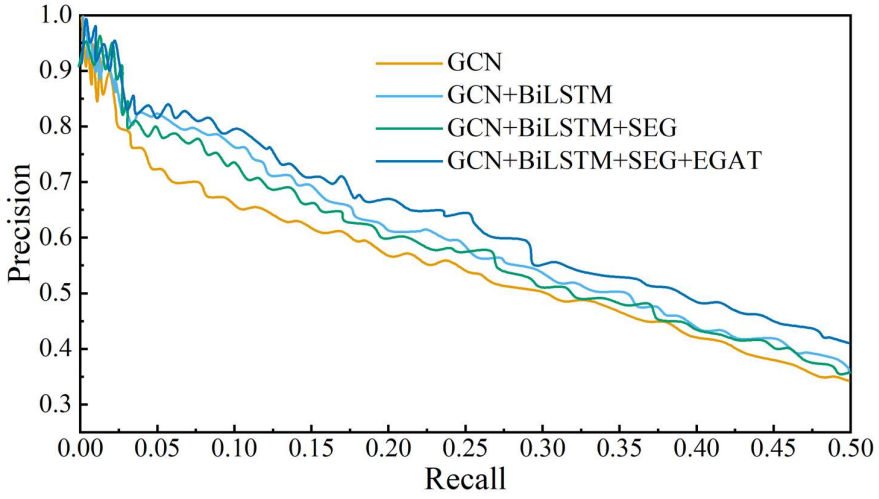

Fig. 6. The experimental results of the experiment

This paper also compares the performance of the proposed algorithm with other benchmark models for processing legal texts on the Law test set. The benchmark models chosen for the traditional approach are the Mintz model, which is the basis of remote supervision, and the feature-based multi-instance learning method (MIML). Where the Mintz model is out classical model for remote supervision and the MIML model is the best model for solving the Mintz problem in the traditional approach. In the deep learning approach, BiLSTM and PCNN methods are used in combination with Attention Mechanism (ATT) and PCNN with Reinforcement Learning (RL) and compared with the proposed method in this paper. Figure 7 shows the test results of different methods on Law dataset.

As can be seen from the figure, the traditional methods Mintz and MIML are much less effective than the deep learning based methods, and the precision drops to 0 when the recall is around 0.25~0.27. The BiLSTM+ATT model has a better precision than the other three methods when the recall is around 0.016, but the precision decreases faster with the increase of the recall. The PCNN+ATT algorithm achieves higher precision on the Law dataset, but its precision advantage is less obvious as the recall increases.The PCNN+RL model outperforms several other algorithms after the recall is greater than 0.15, which also indicates that reinforcement learning can effectively improve the performance of the deep learning relationship extraction model. The precision of this paper's method significantly exceeds that of the PCNN+RL model and several other methods when the recall is 0.01, and its

precision is basically higher than that of other methods after the recall is greater than 0.05. Therefore, the area of the P-R curve of the model proposed in this paper is also larger than that of several other methods when performing legal text entity relationship extraction, thus verifying the superiority of the legal text entity relationship extraction model proposed in this paper.

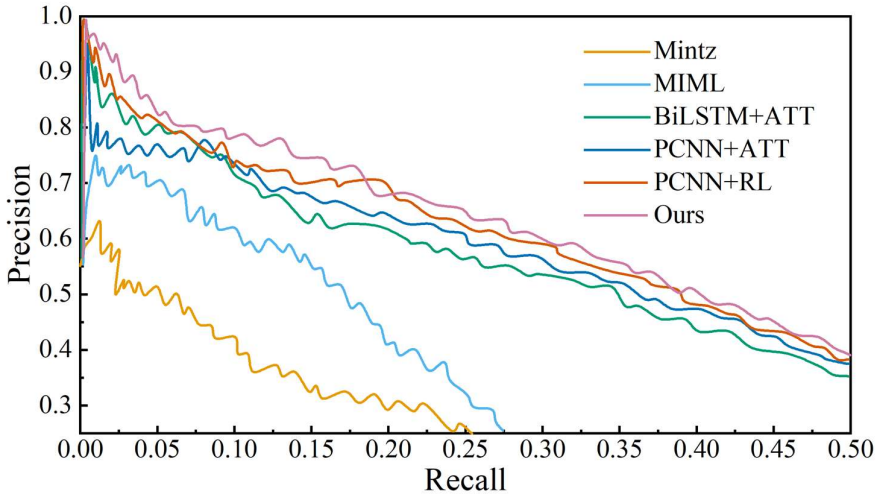

Fig. 7. Test results of different methods

4.2.2 Comparison of overlapping ternary entities. In order to further verify the effectiveness of the GCN-BiLSTM model for extraction on overlapping entities of legal texts, comparison experiments were conducted with CasRel, which is currently more effective in solving entity overlap, using the method proposed in this study in three cases of Normal, Single Entity Overlap (SEO), and Multiple Entity Overlap (EPO), respectively, and the results of the experiments are shown in Table 2.

As can be seen from the table, the accuracy of the GCN-BiLSTM model in the normal type of extraction results is 86.17% and the F1 value is 83.28%, which is not much different from that of CasRel's accuracy of 85.43% and the F1 value of 83.79%, and the accuracy and the F1 value are both improved by less than 1%, which belongs to a slight improvement. This indicates that the two models show very little difference in performance for normal types of legal text entity extraction, which shows that the GCN-BiLSTM model does not improve the performance of normal extraction types very much. However, from the extraction results of single-entity overlapping types, the GCN-BiLSTM model is significantly better than CasRel's extraction results in all the indicators, of which the accuracy, recall and F1 value are 1.42%, 2.02% and 0.78% higher than CasRel's, respectively. It indicates that the GCN-BiLSTM model has a great improvement over the existing model in the type extraction from single-entity overlap, and the extraction effect has a significant improvement. In addition, from the multi-entity overlapping type extraction, the GCN-BiLSTM model has improved both the accuracy and F1 than the experimental results of CasRel. Among them, the accuracy and F1 values are improved by 2.14% and 1.06% respectively over CasRel. It indicates that the GCN-BiLSTM model has a great improvement over the existing model in the type extraction of multi-entity overlap, and the extraction effect has a significant improvement. Taken together, the GCN-BiLSTM model is better than the existing model in solving the single-entity overlap as well as multiple-entity overlap problems in ternary extraction, which proves the feasibility of this kind of extraction effect enhancement of ternary entity overlap by first constructing a legal text graph to represent the relationship, and then extracting the legal text entity relationships under the specific sequential relationship and semantic relationship.

Table 2. Triplet entity overlapping comparison experiment results

Model	Index	Normal	SEO	EPO

CasRel	Accuracy	85.43%	80.27%	79.62%
	Recall	82.26%	79.82%	81.46%
	F1 value	83.79%	80.97%	81.58%
GCN-BiLSTM	Accuracy	86.17%	81.69%	81.76%
	Recall	84.38%	81.84%	80.85%
	F1 value	84.27%	81.75%	82.64%

4.2.3 Cases of substantive relationships in legal texts. This paper focuses on the task of entity relationship extraction of legal texts, for which the extracted set of entity relationship triples can provide data support for the construction of the knowledge graph of legal cases. Through the research work in the previous paper, the entity relationship extraction work of the text of the case domain in the legal field is completed, and in order to better demonstrate the research results of this paper, a case is randomly selected from the Law dataset as an example for illustration. The case has undergone entity recognition, graph representation learning and entity relationship extraction work, a total of 6 ternary groups are extracted, and its entity relationship extraction results are shown in Table 3. The 6 groups of ternary groups extracted from the case, among which ternary group 4 and ternary group 5 belong to a single entity overlapping relationship, and the model proposed in this paper can effectively extract the overlapping relationship, which further validates the effectiveness of the GCN-BiLSTM model. And the legal case text extraction results are visualized, and its visualization results are obtained as shown in Figure 8. Different colors in the figure represent different entity categories, and the relationship between entities is displayed and processed ah through connecting lines, forming entity relationship directed graph, which can more clearly see the association between entities in the legal case text. It is of great help in sorting out the case and further shows the research significance and research value of the work in this paper.

Table 3. Physical relationship extraction

No	Entity 1	Relation	Entity 2
1	Guliang	Has_Offence	Traffic accident
2	Guliang	Has_Light plot	Repentance
3	Traffic accident	Has_According to the rule	Criminal law of the People's Republic of China
4	Guliang	Has_Verdict	Fixed-term imprisonment
5	Guliang	Has_Verdict	Probation
6	Guliang	Other	Liaoning

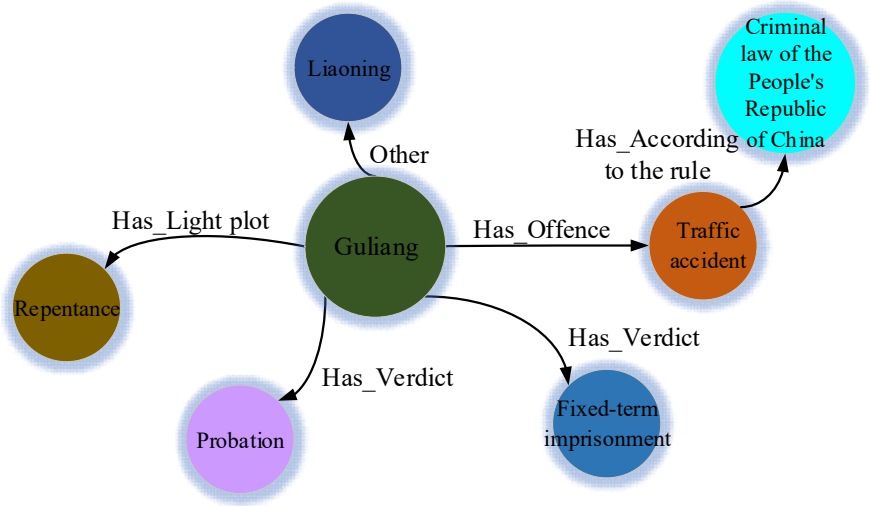

Fig. 8. Physical network visualization

5. Conclusion

In this paper, a GCN-BiLSTM model based on graph representation learning is proposed to be used in legal text entity relationship extraction, which is based on BiLSTM-CRF model for legal text entity recognition, and combines sequential relationship and semantic relationship to construct graph representation attention network. In order to verify the feasibility of GCN-BiLSTM in legal text entity relationship extraction, the article carries out a simulation comparison test. The accuracy, recall, and F1 value of the BILSTM-CRF model constructed in this article are 85.67%, 89.51%, and 88.96%, respectively, when performing legal text entity recognition, which are 7.35%, 8.36%, and 8.32% higher than the single LSTM-CRF model, respectively. The accuracy and F1 of the GCN-BiLSTM model improved by 2.14% and 1.06% over the CasRel model when performing multi-entity overlapping legal text entity relationship extraction. Therefore, applying the graph representation learning algorithm to legal text entity relationship extraction can obtain the entity relationships in legal text and construct a legal knowledge graph based on it, which can provide support for sorting out case information.

References

[1] Hamilton, W. L. (2020). Graph representation learning. Morgan & Claypool Publishers.

[2] Chen, F., Wang, Y. C., Wang, B., & Kuo, C. C. J. (2020). Graph representation learning: a survey. APSIPA Transactions on Signal and Information Processing, 9, e15.

[3] Zadgaonkar, A. V., & Agrawal, A. J. (2021). An overview of information extraction techniques for legal document analysis and processing. International Journal of Electrical & Computer Engineering (2088-8708), 11(6).

[4] Bommarito II, M. J., Katz, D. M., & Detterman, E. M. (2021). LexNLP: Natural language processing and information extraction for legal and regulatory texts. In Research handbook on big data law (pp. 216-227). Edward Elgar Publishing.

[5] Li, R., Li, D., Yang, J., Xiang, F., Ren, H., Jiang, S., & Zhang, L. (2021). Joint extraction of entities and relations via an entity correlated attention neural model. Information Sciences, 581, 179-193.

[6] Zhang, Q., Chen, M., & Liu, L. (2017, December). A review on entity relation extraction. In 2017 second international conference on mechanical, control and computer engineering (ICMCCE) (pp. 178-183). IEEE.

[7] Qin, Y., Yang, W., Wang, K., Huang, R., Tian, F., Ao, S., & Chen, Y. (2021). Entity relation extraction based on entity indicators. Symmetry, 13(4), 539.

[8] Sui, D., Zeng, X., Chen, Y., Liu, K., & Zhao, J. (2023). Joint entity and relation extraction with set prediction networks. IEEE Transactions on Neural Networks and Learning Systems.

[9] Munnelly, G., & Lawless, S. (2018, May). Investigating entity linking in early english legal documents. In Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries (pp. 59-68).

[10] Chen, Y., Sun, Y., Yang, Z., & Lin, H. (2020, December). Joint entity and relation extraction for legal documents with legal feature enhancement. In Proceedings of the 28th International Conference on Computational Linguistics (pp. 1561-1571).

[11] Yin, H., Huang, S., Wang, Z., Ye, Y., & Zhu, W. (2024, August). An Generative Entity Relation Extraction Model Based on UIE for Legal Text. In International Conference on Web Information Systems and Applications (pp. 91-99). Singapore: Springer Nature Singapore.

[12] Liu, D., & Shi, Y. (2024, July). Joint extraction model for overlapping entity relations in legal documents. In Third International Conference on Electronic Information Engineering, Big Data, and Computer Technology (EIBDCT 2024) (Vol. 13181, pp. 1179-1189). SPIE.

[13] Thomas, A., & Sangeetha, S. (2017, October). A legal case ontology for extracting domain-specific entity-relationships from e-judgments. In Proceedings of the Sixth International Conference on Recent Trends in Information Processing & Computing (IPC), Bhopal, India (pp. 27-28).

[14] Li, S., & Yi, L. (2024, January). A Few-Shot Entity Relation Extraction Method in the Legal Domain Based on Large Language Models. In Proceedings of the 2024 Guangdong-Hong Kong-Macao Greater Bay Area International Conference on Digital Economy and Artificial Intelligence (pp. 580-586).

[15] Andrew, J. J. (2018, July). Automatic extraction of entities and relation from legal documents. In Proceedings of the Seventh Named Entities Workshop (pp. 1-8).

[16] Naik, V., Patel, P., & Kannan, R. (2023). Legal entity extraction: An experimental study of ner approach for legal documents. International Journal of Advanced Computer Science and Applications, 14(3).

[17] Enes Celik & Sevinc Ilhan Omurca. (2024). Skip-Gram and Transformer Model for Session-Based Recommendation. Applied Sciences(14),6353-6353.

[18] Asma Mekki,Inès Zribi,Mariem Ellouze & Lamia Hadrich Belguith. (2024). Named Entity Recognition of Tunisian Arabic Using the Bi-LSTM-CRF Model. INTERNATIONAL JOURNAL ON ARTIFICIAL INTELLIGENCE TOOLS(02).

[19] Mazitov D.,Alimova I. & Tutubalina E. (2023). Named Entity Recognition in Russian Using Multi-Task LSTM-CRF. Journal of Mathematical Sciences(4),595-604.

[20] Dexian Wang,Pengfei Zhang,Ping Deng,Qiaofeng Wu,Wei Chen,Tao Jiang... & Tianrui Li. (2024). An autoencoder-like deep NMF representation learning algorithm for clustering. Knowledge-Based Systems112597-112597.

[21] Wang Rui,Xue Linfu,Li Yongsheng,Wang Jianbang,Yan Qun & Ran Xiangjin. (2025). Mineral prospectivity prediction based on the dynamic relation model Atten-GCN: A case study of gold prospecting in the Yingfengjie area, Shaanxi province (northern China). Ore Geology Reviews106399-106399.