Mathematical Biosciences and Engineering

MBE, 20(7): 11585−11604.
DOI: 10.3934/mbe.2023514
Received: 27 December 2022
Revised: 19 April 2023
Accepted: 20 April 2023
Published: 05 May 2023

http://www.aimspress.com/journal/MBE

Research article

A graph neural network-enhanced knowledge graph framework for intelligent analysis of policing cases

Hongqiang Zhu*

Law school, Sias University of Zhengzhou, Zhengzhou 451150, China

* **Correspondence:** Email: 18885@sias.edu.cn.

Abstract: In this paper, we model a knowledge graph based on graph neural networks, conduct an in-depth study on building knowledge graph embeddings for policing cases, and design a graph neural network-enhanced knowledge graph framework. In detail, we use the label propagation algorithm (LPA) to assist the convolutional graph network (GCN) in training the edge weights of the knowledge graph to construct a policing case prediction method. This improves the traditional convolutional neural network from a single-channel network to a multichannel network to accommodate the multiple feature factors of policing cases. In addition, this expands the perceptual field of the convolutional neural network to improve prediction accuracy. The experimental results show that the multichannel convolutional neural network's prediction accuracy can reach 87.7%. To ensure the efficiency of the security case analysis network, an efficient pairwise feature extraction base module is added to enhance the backbone network, which reduces the number of parameters of the whole network and decreases the complexity of operations. We experimentally demonstrate that this method achieves a better balance of efficiency and performance by obtaining approximate results with 53.5% fewer floating-point operations and 70.2% fewer number parameters than its contemporary work.

Keywords: intelligent analysis; policing cases; graph neural network; knowledge graph

1. Introduction

With the high popularity of Informa ionization, individual's work tasks have gradually changed from a manual work mode to an online work mode supported by computer systems. The development of information technology has covered all fields of life. In recent years, computer technology has been applied to the management system of public security cases [1]. General security case information management in the public security bureau has also gradually progressed to Informa ionization; the management of public security cases is critical [2]. In the current management situation, it is necessary to use information technology to improve the efficiency and to reduce the probability of mistakes made by relevant staff in their work, making the management of public security cases more reasonable, standardized, and scientific [3]. In managing general security cases, the public security organs handle the cases in their areas according to their jurisdictions; the possibilities are mainly contained traditionally [4]. Problems regarding the conventional way of managing general security cases includes low efficiency of public security cases, low security of information management, and the need to invest a lot of workforces [5]. The number of cases has significantly increased when compared to the previous years. Additionally, some cases are long, extensive, and complicated, thereby making the traditional way of managing public security cases challenging to meet the current demand for general security case management [6]. Based on research on the level of Informa ionization of case management and the need to build the government ministry's informationized management policy, the Public Security Bureau plans to introduce advanced computer management technology and network technology, etc. [7]. Combined with the demand of the Public Security Bureau for public security case management, there is a need to develop a Public Security Bureau general security case management system to promote the level of Informa ionization of the Public Security Bureau for public security case management [8].

The concept of big data is flourishing globally, and the core value of its technology is to provide a complete reference for the development of the situation with real predictive power [9]. When big data is immersed in all walks of life, and the country dramatically supports the development of big data, the public security industry keeps pace with the technology of the times, and the general security industry has its special status. People have always despised and rejected crime, and the public security industry dramatically contributes to combating such activities [10]. Each case may be related to a large amount of data, and how to grasp and reason out the helpful information from a large amount of data is a test for the public security industry in the current era of big data. The most basic application of big data technology in general security is to build a big data management platform. The most current application of big data in public safety is constructing a data management platform [11]. When the volume of data becomes large, manual screening becomes powerless, and the accuracy and efficiency drops, thereby increasing the urgency for data cleaning and data mining for police intelligence analysis in the current era of big data. Public security constantly deals with crime, as all general security data are mostly related to people, and a user profile is a collection of tags [12]. It is difficult to establish a user profile because it requires a large amount of data scattered in various systems, and it is impossible to determine which data belongs to the same person. In addition, the large amount of personal information brings technical difficulties to storage and query. The knowledge graph is good at handling data with complex entity relationships, so the research on user portraits based on the public security knowledge graph can solve the two aforementioned problems.

This paper uses Graph Neural Network (GNN) models to establish high-level associations between complex features and to accomplish relational reasoning and information interaction, thus avoiding the adverse effects of direct fusion of multimodal or complex feature representations [13]. A graph model is an approach proposed for non-Euclidean spatial data. Researchers have recently borrowed ideas from convolutional networks, recurrent networks, and deep autoencoders to extend graph models specialized for non-Euclidean spatial data to form neural networks in deep learning models. The theory of immobile points was the theoretical basis for graph neural networks when they were proposed [14]. Specifically, given a graphical model in which an initial state characterizes each node, the learning goal of a graph neural network is to obtain the hidden forms of all nodes by integrating the information of neighboring nodes [15]. According to Banach's immobility point theorem, the iterative update converges to a dormant state if the global update function implemented by the feedforward neural network is guaranteed to be a compressed mapping. Researchers have recently proposed utilizing a gating-based information integration and a state update method to iterate fixed steps to obtain the desired information interaction and integration results more efficiently [16]. The researchers found that the model can effectively handle complex data, especially relational reasoning and information interaction, with unparalleled advantages. This has made it increasingly popular in various fields, such as social networks, knowledge graphs, recommender systems, and the life sciences. Since graph neural networks are an emerging research result, their application in computer vision is still exploratory [17]. Therefore, in this paper, graph neural networks are exploratively applied to scene understanding tasks to solve the problem of information interaction and relational inference of complex features in convolutional neural networks. In practical application scenarios, the difficulty of multimodal, multiscale, and multistage feature fusion of various expressive features can better verify the adaptability and stability of the algorithm. In this paper, related work is introduced in Section II. In Section III, the graph neural network-enhanced knowledge graph framework based on intelligent analysis of policing cases is studied in depth. A knowledge graph based on a graph neural network is constructed to build a knowledge graph embedded for policing issues. Additionally, a graph neural network-enhanced knowledge graph framework based on intelligent analysis of policing issues is designed. Section IV tests the knowledge graph model of neural graph networks and implements the graph neural network-enhanced knowledge graph framework for thoughtful analysis of policing cases. Section V concludes that experiments demonstrate that the method proposed in this paper achieves a high accuracy rate in case similarity calculation.

2. Materials and methods

In the 1980s, the Crime Information System, the Population Information System and the Exit-Entry Management Information System were built, and these systems played a significant role at that time. In the 1990s, public security departments paid increasing amounts of attention to construct information technology [18]. They built the National Crime Information Centre (CCIC), a nationwide crime information center built by the Ministry of Public Security [19]. It is a fast collision response system supported by the existing public security computer information system and network [18]. In this century, the general security departments have unified their understanding, and the development of information technology has become faster and faster, especially the construction and promotion of the "Golden Shield Project", which provides a good network foundation for handling cases online [20]. Public security organs at all levels have websites, and many things can be done online [21]. The information construction of general security departments has made substantial progress, and the whole department is developing. In addition to the government departments vigorously promoting information technology building, many enterprises have also started the development of public security information software.

Handling public security cases is the most basic and frequent business work of the general security police, which is essential for maintaining social stability and protecting citizens' life and property [22].

In traditional professional teaching in public security colleges, the education of handling general security cases can only be completed through teachers' lectures on theory, in which students analyze cases, watch video materials, and observe scenarios in actual departments, etc. There is a lack of effective practical teaching carriers and modes, in addition to an urgent need for a safe, efficient, intuitive, and reliable solution to simulate the actual work of public security organs in handling general security cases using computer information technology [23]. The critical public security's practical teaching is safety, efficiency, intuition, and reliability. The core of neural networks is to iteratively pass and enhance the feature representation of nodes by propagating information between nodes and incorporating microscopic aggregation and update functions [24]. V. Hassija et al. first proposed the concept of neural graph networks, which extends recurrent neural networks to handle graphical structure data [25]. K. H. de Jesus Prado et al. proposed using graph convolutional networks to capture the relationships between objects in video recognition tasks [26]. C. Jin et al. used graph convolutional networks to explain the connections between different regions in scene analysis [27]. Since most models can achieve similar performances with sufficient data, zero- and few-time learning are gaining importance in image classification. Much work has started investigating how structural information can be integrated into image classification using graph neural networks. First, the knowledge graph can be used as additional information to guide the zero-recognition type. P. M. Asaro designed a knowledge graph in which each node corresponds to a category and predicts different categories using the word embeddings of the nodes as inputs to the classifier [28]. Since the deep part of the convolutional network produces over-smoothing effects, one can solve this problem by utilizing the propagation of graph convolutional neural networks. M. A. AlGhamdi and M. A. Khan used a single-layer graph convolution module with a larger neighborhood that included adjacent and multi-step nodes in the graph [29]. This method is experimentally effective in combination with existing zero-order classifiers.

Knowledge graph technology includes creating and applying knowledge graphs, which involves interdisciplinary disciplines such as the semantic Web, natural language processing, and machine learning [30]. Regarding the implementation process, knowledge graph technologies can be divided into knowledge graph construction technologies, query and inference technologies, and knowledge graph applications. In the following, the knowledge graph construction techniques are highlighted. The process of knowledge graph construction can be divided into three steps: knowledge acquisition, knowledge fusion, and knowledge storage [31]. Entity, relationship, and event extraction are usually included in knowledge acquisition. Entity extraction is also called named entity recognition, and there are mainly two types of recognition methods: rule-based entity recognition methods and machine learning-based entity recognition methods [32]. In the early stage of named entity recognition research, rule-based methods dominate, especially the technique based on the named entity dictionary is the most representative. The method achieves faster testing results on small-scale corpora in specific domains. However, this process requires linguistic experts to write rules artificially, requiring high language knowledge, many human and material resources, poor portability, and low generality. Police video investigations mainly study how to extract the knowledge model of video investigation and the thinking model of information analysis from many cases to realize the effective management and application of multimedia information and knowledge of patients. Therefore, the knowledge model extracted from the investigation cases is the concentrated essence of various excellent investigative thinking, the methodology when using video investigation means for case detection, and the reference basis for managing and applying the video information involved in the case [33]. Knowledge mapping and related technologies can perfectly integrate the extraction and application of such models, establish a structured semantic knowledge base, and realize the complete mining of video data. Therefore, video investigation can play certain advantages in public security combat and provide the maximum number of clues to case officers.

3. A graph neural network enhanced knowledge graph framework based on intelligent analysis of policing cases

This paper serves as data support for domain applications by the need to construct domain knowledge graphs. This chapter proposes a system for building domain knowledge graphs and designs a graph neural network-enhanced knowledge graph framework based on intelligent analysis of policing cases, as shown in Figure 1.

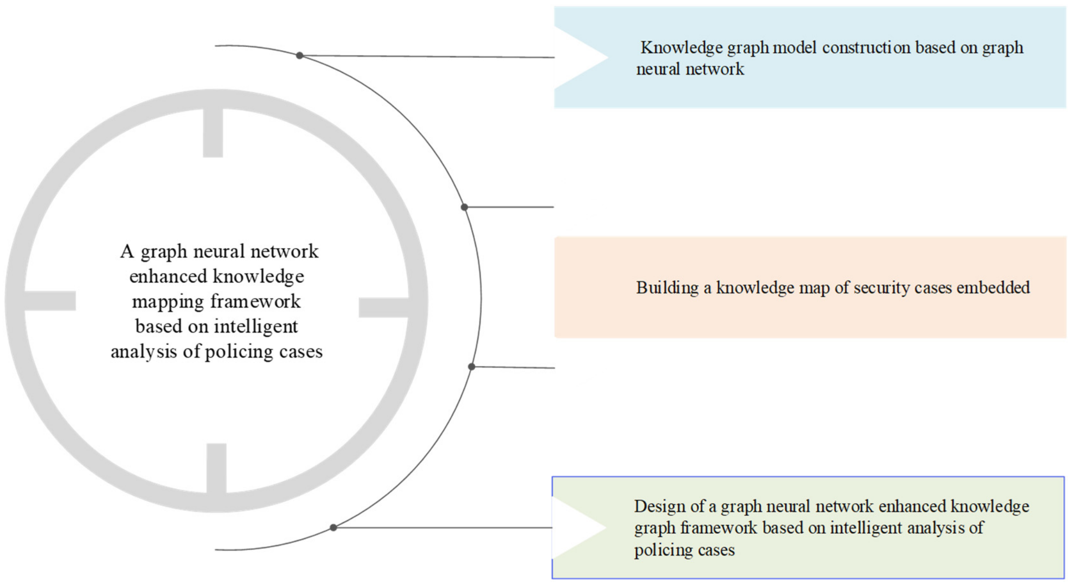

Figure 1. Framework overview.

3.1. Knowledge graph model construction based on graph neural network

As an extension of neural networks on graph-structured data, a Graph Convolutional Network (GCN) is designed for irregular graph-structured data by borrowing the ideas of traditional neural networks. The graph neural network is based on the message-passing mechanism to learn the embedding representation of nodes in the graph, i.e., each node in the chart is updated by aggregating the embedding representation of neighboring nodes. The nodes in the graph can continuously receive information from more distant nodes as the training progresses to update the embedding representation of nodes constantly. Different types of neural chart networks are generated depending on the aggregation strategy of neighbor nodes. The graph convolutional neural network GCN is a classical graph neural network model that differs from the traditional graph neural network. In addition to the message-passing mechanism that summarizes the working principle of GCN, it can also be compared to the graph neural network based on the graphical structure using the convolution operation [34]. Compared to the previous graph neural network model under the convolution operation, GCN maintains the model effect while reducing the model effect by retaining only the maximum. The complexity of the model parameters is reduced by having only the maximum eigenvalues, which allows GCNs to be used in various applications of graph-structured data. When using the graph convolutional neural network for learning the embedding node representation in the chart, the embedding node representation h_i^l of the graph node i at the l layer can be calculated by Eq (3.1).

$$h_i^l = w^l - h_j^{(l+1)} + f\left(n_1(i) + \sqrt{\sigma \times c_i}\right). \tag{3.1}$$

where $n_1(i)$ denotes the set of first-order neighbor nodes of a given node i, w^l denotes the layer's weight matrix in the graph neural network, c_i denotes the normalization constant, and $\sigma(\cdot)$ denotes the activation function. The node vector of the input layer is represented by h_i^o. The GCN uses average pooling to aggregate the embedding representation of all neighboring nodes when performing the message aggregation operation, which means that all adjacent nodes play the same role in the learning process of the embedding representation for the central node. The overall flow of the graph neural network is shown in Figure 2.

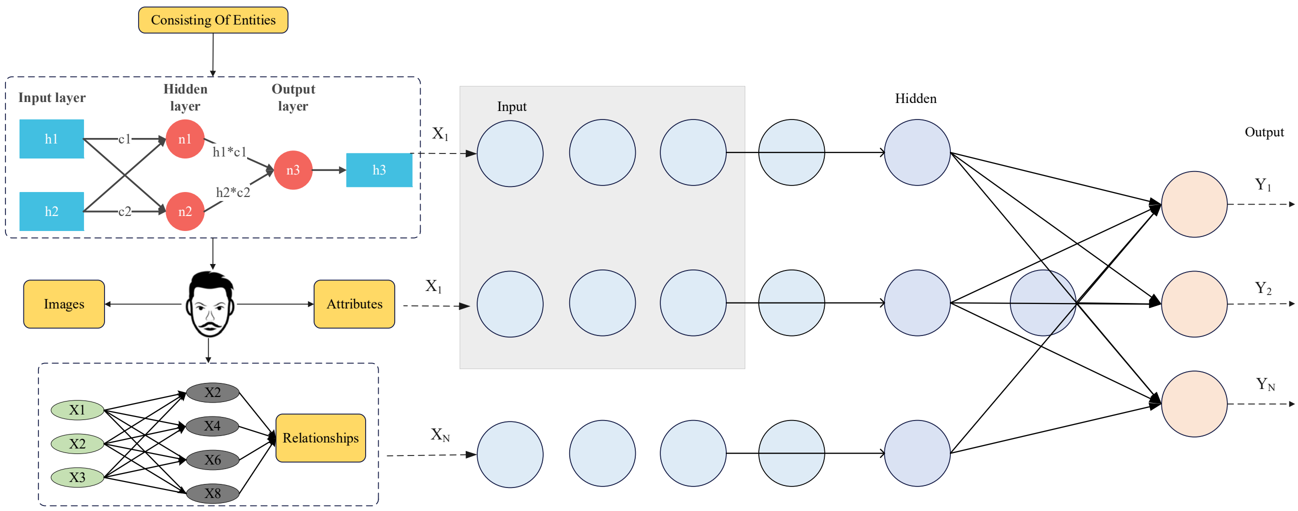

Figure 2. The overall flow of graph neural network.

A knowledge graph is a digital structure representing knowledge as concepts and relationships between them, consisting of entities, images, attributes, and relationships. There are three ways to construct a knowledge graph: bottom-up, top-down, and a mixture.

1) Bottom-up: The process of constructing a knowledge graph from the bottom-up includes, first, extracting knowledge from open multi-source data using machine learning techniques such as named entity identification and relationship extraction, then normalizing and integrating knowledge from different sources using techniques such as entity alignment and attribute value filling, and finally performing concept abstraction and quality assessment on the constructed data layer.

2) Top-down: The specific steps of the top-down knowledge graph construction method are to

build a well-structured concept hierarchy tree starting from the topmost layer using ontology learning or rule definition and then to populate the entities obtained from knowledge extraction into the schema layer ontology by techniques such as entity linking and entity filling. With the demanding high quality and accuracy requirements in a specific domain, domain knowledge mapping develops a complete ontology layer schema first. Thus, it usually adopts a combination of bottom-up and top-down construction methods. In constructing knowledge graphs, the obtained knowledge needs to be represented. The knowledge representation of knowledge graphs has been completely shifted from a symbolic logic-based approach to a semantic web-based system and a knowledge graph representation learning approach. Semantic Web is one of the most widely used approaches for knowledge representation of knowledge graphs, which utilizes nodes to represent concepts, edges to illustrate relationships, and a mesh data structure formed by nodes and edges connecting nodes to create a graph to represent complex knowledge graphs. The Trans E model minimizes the following edge-based loss functions to learn these embedding representations of entities and relationships on the training set.

$$l = \sum_{s=1} (h,r,t) \frac{\gamma(h-r\times t)}{[X]+d(x,y)}. \tag{3.2}$$

where [X]+ denotes the positive part of x, $\gamma > 0$ denotes the margin hyperparameter, and d (x, y) denotes the calculated distance between the two vectors of x and y.

$$s_h = \sum \frac{(h,r,t)\times(t+\gamma)}{\sqrt{(h,r,t)\times(h-\gamma)}}. \tag{3.3}$$

Equation (3.3) represents the negative sample set formed when one of either the head or tail entities in the correct triad is replaced by one of the random entities in the entity set.

In this paper, after initializing the embedding node vectors, the information propagation layer iteratively aggregates the neighborhood node embedding vector information. It updates the embedding vectors of the optimized target nodes for information propagation. The information propagation layer in the proposed algorithm consists of two network layer structures, the knowledge graph-enhanced bilinear propagation layer and the knowledge graph-enhanced linear propagation layer. The difference between the two network layer structures is that the aggregation method adopts the bilinear. The difference between the two network layer structures is that the aggregation method adopts the bilinear and linear aggregation methods.

The information propagation process in graph neural networks is divided into three phases: information construction, information aggregation, and representation update. This subsection focuses on the information aggregation phase's linear and bilinear aggregation problems. Most existing graph neural networks, such as GCN and Graph Attention Network, use weighted sums to aggregate the information of the target node's neighboring nodes, also called linear aggregation. Although the linear aggregation of neighbor nodes results in aggregating data in a variety of neural graph networks, the aggregation approach, which cannot capture the possible interactions between neighbor nodes of the same target node, limits the expressiveness of neural graph networks using the linear aggregation approach when the interactions between neighbor nodes can serve as an essential signal to improve the performance of the algorithm. The bilinear aggregation approach is inspired by factorization machine-like models such as Factorization Machines (FM) and Neural Factorization Machines (NFM). It uses a combination of second-order node embedding vectors to model the interactions between pairs of neighboring nodes of the target node, as computed in Eq (3.4).

$$h_{n(v)} = \sum_{i,j=1}^{n} \frac{\frac{1}{2}[n(v)]*[n(v)-1]}{w\left(v_i - v_j\right)}. \tag{3.4}$$

In Eq (3.4), v denotes the target node to be information aggregated, $h_{n(v)}$ denotes the embedding vector of the target node v after aggregating the neighborhood information, i and j denote the serial numbers of the nodes in the set of neighborhood nodes of the target node v, v_i and v_j denote the embedding vectors of node i and node j, respectively, and $\frac{1}{2}[n(v)]*[n(v)-1]$ denotes the regularization coefficients to eliminate the effect of different node degrees. The bilinear aggregation approach models the possible interactions between pairs of neighboring nodes by summing the elemental products of the embedding vectors of pairs of nodes while aggregating the neighboring node information of the target node.

Linear aggregation and bilinear aggregation are two different approaches in the node information aggregation stage, and each of them display various advantages depending on the graph structure data. The bilinear aggregation approach can achieve better aggregation results when there is a mutual influence relationship between the neighboring nodes of the target node. In collaborative knowledge graph, user nodes only interact with product nodes, and their information aggregation update process explicitly aggregates product node information. Meanwhile, information of attribute entity nodes can be implicitly propagated to user nodes through product nodes through information propagation. There is a certain similarity between product nodes adjacent to the same user node, so bilinear aggregation is used to aggregate the information of product nodes and update the embedding vector of user nodes.

3.2. Building a knowledge map of policing cases embedded

In this paper, we sample Trans E to embed the set S; the basic assumption of Trans E is the displacement assumption, i.e., the head entity vector plus the relationship vector obtained after the knowledge graph embedding is completed should be approximately equal to the tail entity vector. So, it applies to the triples in the knowledge graph, mostly one-to-one relations. At the same time, its variant Trans H can achieve one-to-many relation representation by projecting the links onto a hyperplane. While other variants, such as Trans R, achieve a more robust picture of entities through a hypothetical semantic space, Trans D uses two vectors for each entity and relationship to enhance its representation capability. In this paper, we adopt the following rules to convert a knowledge graph into an undirected graph: the entities in each triple (h, r, t) of the knowledge graph are converted into a node, respectively, and the number of the node corresponds to the entity name; the relations in each triple are converted into a node, and the number of the node is incremented in turn. The embedding vectors of the entities or relations are used as the features of the nodes. Such a conversion can realize that the converted graph has the same topology as the original knowledge graph and can fully utilize the embedding representation of the knowledge graph [35]. In the vertical domain knowledge graph constructed in this paper, the relationships in the triad are set as undirected graphs because the direction does not have a significant effect.

This paper proposes a similarity calculation method based on graph convolution (GCN) and bi-directional long and short-term memory (Bi LSTM) network with the network structure shown in Figure 3. The network structure contains two modules: the embedding module, which is used to learn the representations of the input graph, and the similarity calculation module, which calculates the similarity between the case graph representations. The embedding module contains two components: the GCN network layer applies the graph convolution operation to update the node features, and the

Bi LSTM network layer aggregates the node features to obtain the parts of the whole graph. During training, three cases for each sample and their charts go through the same embedding module to get the final representation.

Like other ranked similarity learning tasks, the case similarity calculation problem, for a given case triple of sample i$\left[a^i,b^i,c^i\right]$, where the similarity of cases a^i b^i is greater than the similarity of cases a^i c^i, $\left[a^i,b^i\right]$ is called the cheerful pair and $\left[a^i,c^i\right]$ is the opposing pair. The $\left[e_a^i,e_b^i,e_c^i\right]$ Graphs $\left[g_a,g_b,g_c\right]$ representing the three cases in sample i are passed through the above embedding module to obtain the embedding vector. To learn the parameters of the whole network Θ, this paper defines the loss function of the network as in Eq (3.5):

$$L=\sum_{i=1}\max\frac{d\left(e_a^i-e_c^i\right)}{d\left(e_a^i-e_b^i\right)}\times m. \tag{3.5}$$

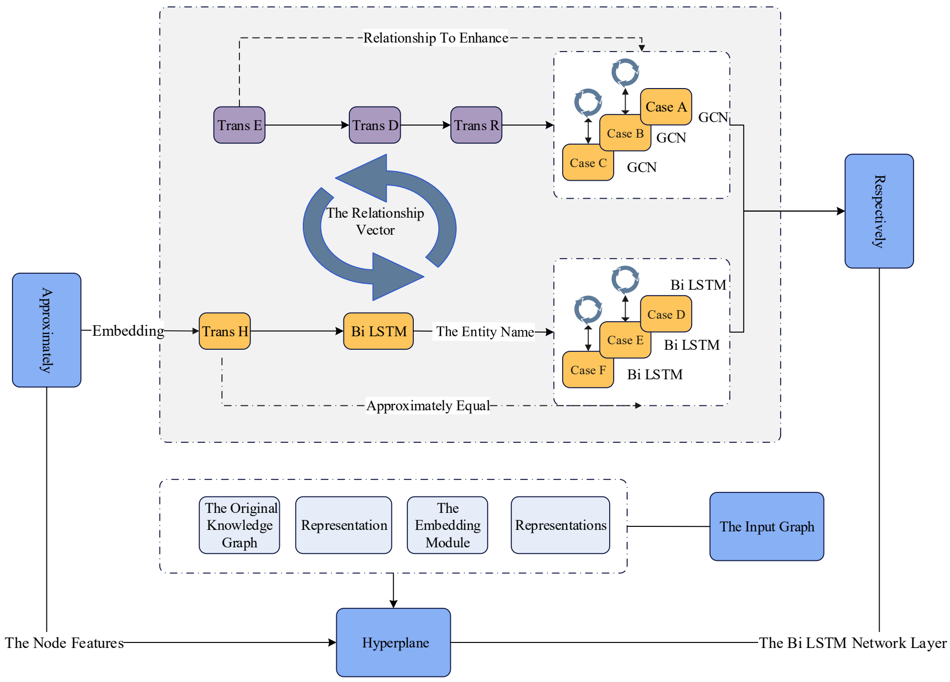

Figure 3. The overall structure of the network model is based on GCN and Bi LSTM.

In this paper, we propose to utilize the Bi LSTM network layer to dynamically aggregate the eigen information of each node in the graph. Given the output $H^L=\left[h_1,h_2,\cdots,h_n\right]$ of the L layer of the graph convolution, $h_i\in R^d$, which denotes the hidden vector of the i node in the graph after the stacked graph convolution layer, the remote vector of each node is used as the input vector of each cell of the Bi LSTM, and the t cell unidirectional is calculated as follows:

$$
\begin{aligned}
c_t &= \tanh w_c\left[a_{t+1}, h_t - b_c\right] \\
u_t &= \sum w_u\left[a_{t+1}, h_t - b_u\right] \\
f_t &= \sum w_f\left[a_{t+1}, h_t - b_f\right] \\
o_t &= \sum w_o\left[a_{t+1}, h_t - b_o\right] \\
c_t &= \sum u_t\left[c_t + f_t \times c_{t+1}\right] \\
a_t &= o_t \times \tanh\left(c_t\right)
\end{aligned}
. \tag{3.6}
$$

Where a_t is the hidden state of the t cell unit, c_t is the cell state of the t cell unit, and c_t, u_t, f_t, o_t is the cell gate, update gate, forget gate, and output gate, respectively.

3.3. Design of a graph neural network enhanced knowledge graph framework based on intelligent analysis of policing cases

The system mainly includes data acquisition and storage modules, entity extraction, relationship extraction, summary generation, knowledge retrieval, and visualization. The data acquisition and storage module refer to crawling, parsing, and storing the data of Baidu Encyclopaedia entries. The entity extraction module refers to extracting entities from text. The relationship extraction module extracts entities, relationships, and attributes in phrases based on dependency syntax analysis. The summary generation module removes text summaries based on the Text Rank algorithm and sentence synthesis similarity. The knowledge retrieval and visualization module retrieves entities, relations, and visualization from the Neo4j database. The input image is cropped with 2x magnification, and the local training network is trained on the two parts of the cropped image by inputting them to the upper and lower branches, respectively. The residual network method is learned while improving the training learning rate and removing obstacles for gradient transmission. Feature extraction is performed and combined at the local training network's end; the result is the reconstructed high-resolution face image. The regional training is divided into three main parts: image cropping, feature extraction, and face replacement. The structure diagram of the local training network is shown in Figure 4.

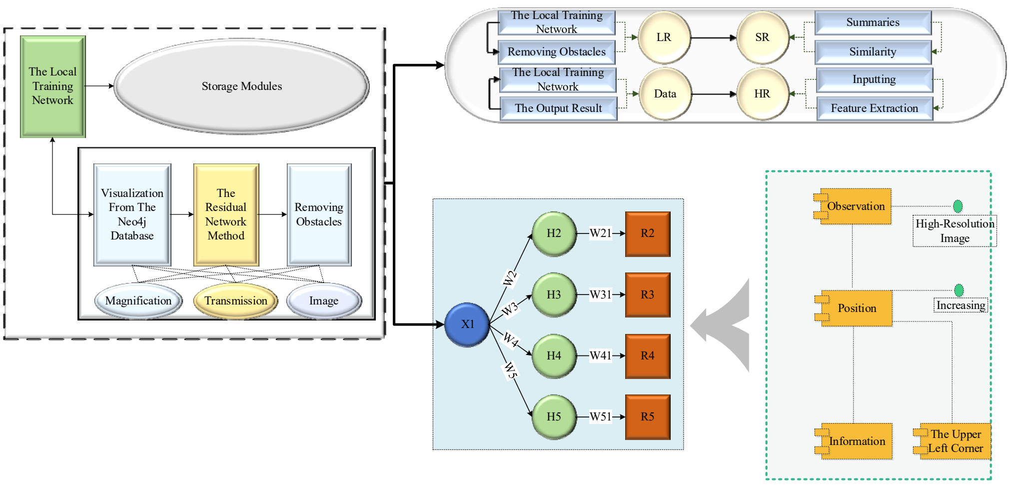

Figure 4. Structure of local training network.

In the error calculation of the reconstructed super-resolution image (SR) output from the neural network with the original high-resolution image (HR), this chapter takes the operation of increasing the error weight of the face region and decreasing the error weight of the background region. The face region is selected from the observation that the eyes, nose, and mouth of the face are in the same position because the input low-resolution image (LR) is aligned. Based on this prior information, the beginning is considered in a square region with the upper left corner of the reconstructed SR image as the origin, with pixel location (25, 25) and width and height of 80, the loss of the face region is Eq (3.7).

$$l_{facial} = \sum_{i=25}^{105} SR\left(i-j\right) + \sum_{j=25}^{105} \frac{HR(i+j)}{80 \times 80}. \tag{3.7}$$

Knowing the loss value of the whole image and the loss value of the face region, the loss value of the background region is the total loss value minus the loss of the face region in Eq (3.8).

$$l_{backgound} = \sum_{i=0}^{127} \frac{SR(i-j)}{128 \times 128} + \sum_{j=0}^{127} HR(i-j). \tag{3.8}$$

The image super-resolution reconstruction algorithm aims to obtain a mapping function to achieve a high-resolution image output from an input low-resolution image. The quality of image reconstruction is measured by subjective evaluation and objective evaluation. The common practice of personal assessment is to gather a group of experimenters and give a subjective rating to the reconstructed images. The statistical rating is used as the score of the image reconstruction quality. This practice is influenced by the experimenter's and other environmental factors, and the evaluation results are subject to significant errors. Therefore, this section uses objective evaluation criteria to measure the quality of image reconstruction. The commonly used evaluation criteria include peak signal-to-noise ratio (PSNR) and mean squared error (MSE). PSNR indicates the percentage of maximum possible power to noise intensity. A more considerable PSNR value between two images means they are more similar, while the opposite scenario signifies less similar values. Since the photos are stored discretely as pixel values, the maximum possible power can be used as the maximum pixel value in the image. The formula is given in Eq (3.9), where MAX is the maximum gray value in the picture, usually 255.

$$l_{(i_{sr}-i_{hr})} = \sum_{i=j} MSE \frac{SR(i-j)}{HR(i-j)}. \tag{3.9}$$

4. Results and analysis

4.1. Knowledge graph model testing of graph neural network

Design of a graph neural network enhanced knowledge graph framework for intelligent analysis of policing cases according to the previous section. This paper selects a single model as an additional autoregressive multichannel graph neural network model. For predicting policing case volume, the first step is to find the factors associated with causing the occurrence of community policing risk. Since convolution has the role of local feature extraction, in this paper, feature extraction is essential for analyzing community policing case distinguishing factors. According to the need for policing case prediction, this section improves the traditional graph neural network while using multiple feature factors as the input values of the graph neural network. It proposes an autoregressive multichannel graph neural network model to predict the policing case volume. The web consists of the input, convolutional, top pooling, flatten, dropout, and fully connected layers. The Dense Net is added after the fully connected layer; the Dense Net can establish the connection between different layers and fully use the features to make each Feature Map connected, which can alleviate the problem of gradient disappearance. In this paper, the graphical neural network is improved by replacing the traditional visual neural network with a multichannel prediction model, considering that multiple feature attribute impacts the prediction of the number of cases. The improvement enhances the graph neural network's perceptual field and the model's prediction accuracy. The comparison of the graphical neural network data test before and after the modification is shown in Figure 5.

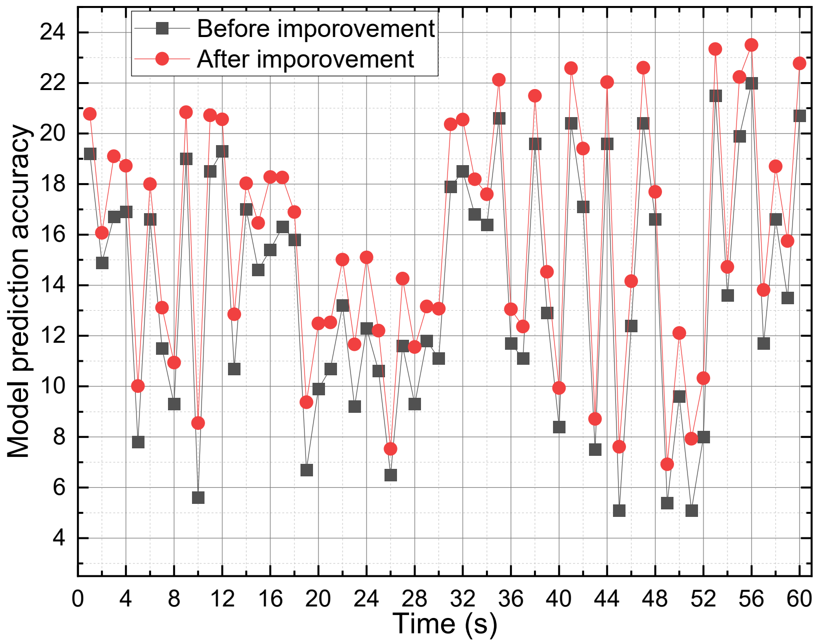

Figure 5. Comparison of graph neural network data testing before and after improvement.

To verify the effectiveness of GCN and Bi LSTM, this paper compares the accuracy changes of the models under different network structures and sets up the models with the following configurations for comparison: 1) to verify the effect of GCN, the GCN layer in the network is removed, and the node feature information matrix is directly input into the Bi LSTM layer to form the Bi LSTM model output; 2) to verify the effect of Bi LSTM layer, the Bi LSTM layer after the GCN layer is replaced with the global max pooling (GMP) layer to form the GCN+ GMP model output; and 3) to further compare the graph pooling effect, the GCN layer is replaced with the global max pooling (GMP) layer to form the GCN+ GMP model output. To verify the effect of the Bi LSTM layer, one can replace the Bi LSTM layer after the GCN layer with Global max pooling (GMP) to form GCN+ GMP model output; to further compare the effect of graph pooling, one can replace the Bi LSTM layer after GCN layer with Global average pooling (GAP) to form GCN+ GAP model output; to verify the effect of Bi LSTM layer, one can replace the Bi LSTM layer after GCN layer with Global average pooling (GAP) to form GCN+ GAP model output. GCN+ GAP model output; the model's accuracy on the validation set is obtained when each model is trained for the same 200 cycles (epoch), at which time the models have converged, and the accuracy rates of different network structures are shown in Table 1.

Table 1. Accuracy rates of different network structures.

Network Structure	Training value	Accuracy
Bi LSTM	0.378404464	58.35%
GCN	0.01722239	61.65%
GMP	0.788480225	65.73%
GAP	0.663970796	67.87%
GCN+GMP	0.449490187	74.78%
GCN+GAP	0.752821286	76.21%
GCN+Bi LSTM	0.166160123	83.57%

The traditional network is changed from a single channel to four feature inputs: Inputs1, Inputs2, Inputs3, and Inputs4; below each input feature are two convolutional layers; below each channel convolutional layer are Max Pooling1, Max Pooling2, Max Pooling3. The next is the fully connected layer Connected; the following is the Dense Net, where each constituent layer uses BN and Re LU, and then the output features of the four channels are mapped for 3 × 3 convolution, and the Dense block is three layers. The Dense block is a three-layer convolutional network; an average pooling follows the Dense block, and a SoftMax classifier is added. Then a dropout layer is connected, and the dropout is 0.5. Finally, an output layer is added, and the output result is the prediction result value of the multichannel convolutional neural network, which is the number of policing cases in this paper.

For each dataset, 70% is chosen as the training set, and the remaining 30% is the test set in this chapter. The parameter settings are first based on the original text for the comparison model to ensure a fair comparison. On this basis, the comparison model is optimized as much as possible. For the Knowledge Graph Attention Network (KGAT) model, the parameters are set as follows for each dataset: the learning rate and batch size are set to 0.02 and 2048, respectively, for the Movielens-1M dataset; the learning rate and batch size are set to 0.0005 and 128, respectively, for the Last FM dataset; the learning rate and batch size are set to 0.0005 and 128, respectively, for the Book-Crossing dataset. In addition, the number of neighbors and hops are 4 and 2 for the Movielens-1M dataset and 8 and 1 for the Last FM and Book-Crossing datasets, respectively. The default aggregator is the BI-Interaction aggregator. In addition, an early termination policy is implemented if HR@20 and NDCG@20 do not increase for 20 consecutive epochs on the test set. The analysis of the experimental results for the comparison model is shown in Figure 6.

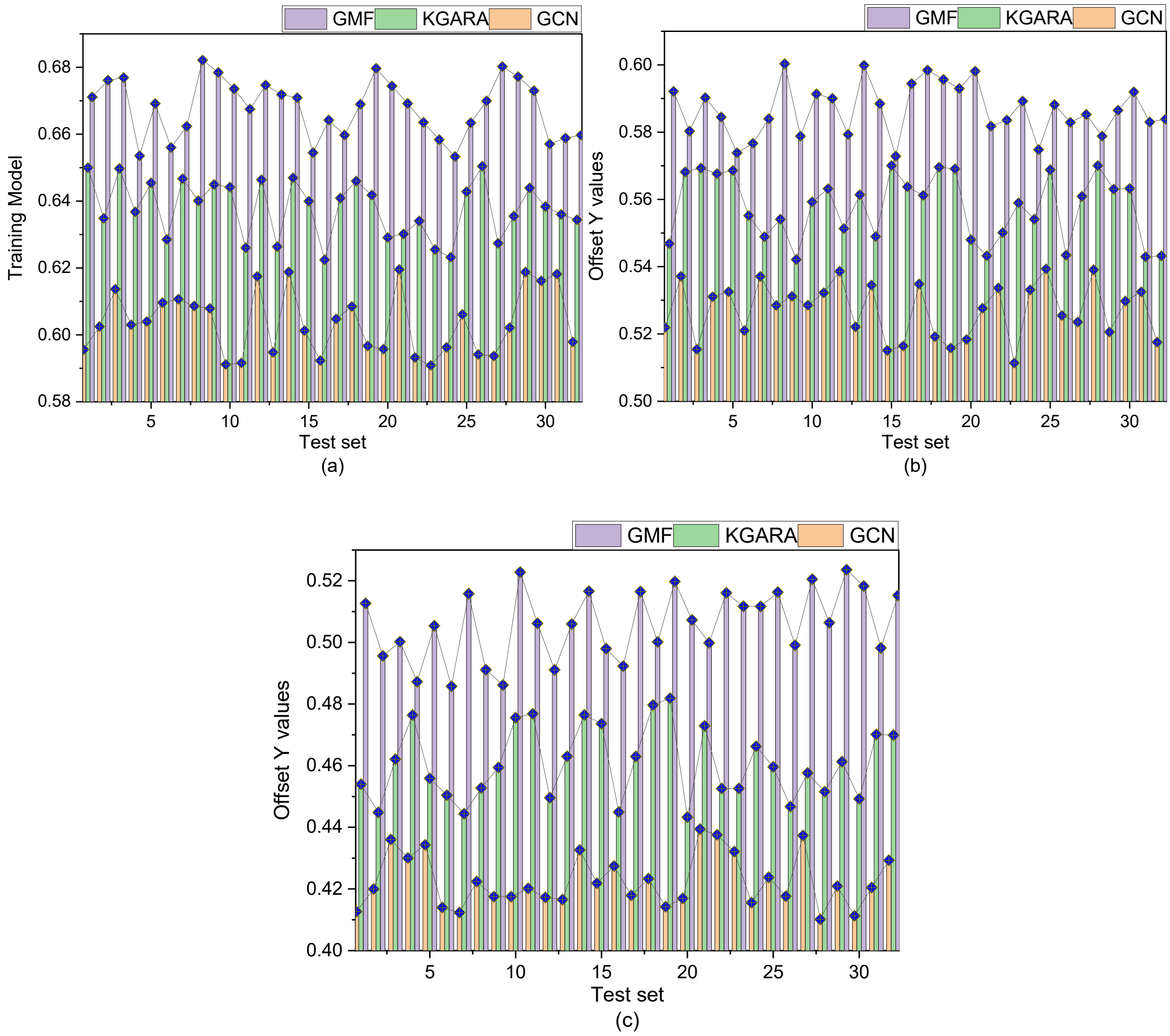

Figure 6. Analysis of experimental results of the comparison model.

4.2. Implementation of a graph neural network-enhanced knowledge graph framework for intelligent analysis of security cases

The Knowledge Graph Retrieval and Visualization module retrieves nodes and relations from the Neo4j database, retrieves the database using Cypher search statements, or uses the visualization interface of the Neo4j database. This module contains four elements: node retrieval, relationship retrieval, complex retrieval, and Neo4j graph database visualization. Node retrieval refers to entering the code name and returning the base information of the node. In this paper, we compare the effect of window size on the model through experiments. The experimental results are shown in Figure 7. The results show that the impact of named entity recognition gradually improves with the increase in window size. When the window size is 7~11, the recognition effect is close, and all of them are above 97%. Because the rise in the window will increase the number of parameters of the model and lead to a significant rise in the model training time, the window size of 7 is used in the later experiments of this paper after careful consideration.

The model uses a multi-task learning strategy in the training process. One of the main tasks is the sequence recommendation task, which optimizes the sequence recommendation effect of the model based on the model's prediction and actual results. The contrast learning task is used as an attribute task, and the construction depends only on the sequence of items input to the model and the construction process of the contrast learning task in the dotted line. First, other data enhancement methods score two things with different perspectives for the same sequence of items. Then, the two sequences are used as inputs to the model to obtain a representation of the sequences. Finally, the consistency of the two terms is maximized by comparing the constraints of the loss functions. It is worth noting that the two sequences obtained from the same sequence by data augmentation are positive sample pairs. The overall goal of the task is to keep the information of different perspectives of the same sequence consistent, i.e., the positive sample representations remain consistent.

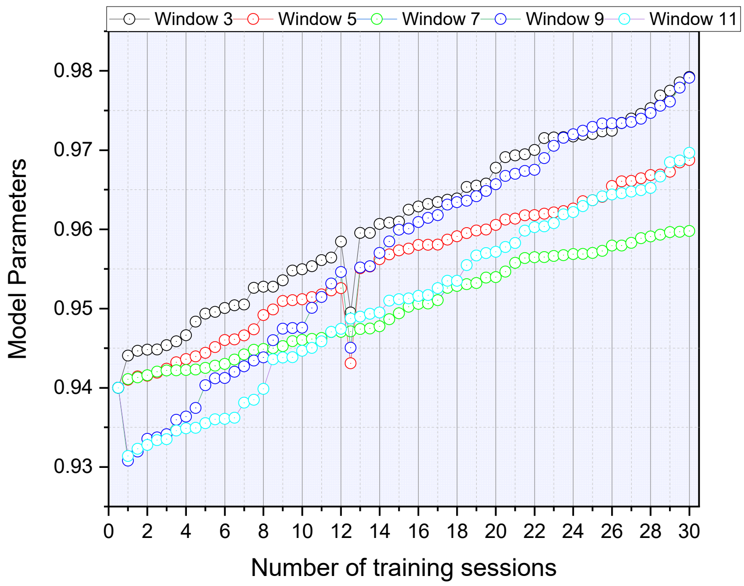

Figure 7. Temporal data variation of model training.

According to the model training using the text extensive data analysis in the knowledge graph, by analyzing the features of the case text, semantic deep learning technology systems can realize the function of automatic classification according to the case content. If the public security case data is imported into the program, the program will automatically classify the cases. Based on the refinement of case classification, it can realize the monitoring of case trends and improve the control of points for a short time. At the same time, knowledge mapping technology is used to analyze textual information such as case descriptions and transcript records to refine the modus operandi, means, and other characteristics of multiple cases and, accordingly, achieve the ability to analyze different types of cases in series.

This paper analyses the effect of various data enhancement operations on the experimental results of case analysis. This paper uses both insertion and replacement methods for short sequences and long lines; this paper changes the set of optional plans to perform the experiments. The effects of the network-enhanced knowledge graph on case analysis are shown in Figure 8. The variables s, i, m, r, and c denote replacement, insertion, masking, reordering, and cropping, respectively. The model's effectiveness decreases significantly when the data enhancement method is SI or SIR. This is due to the overfitting of the model caused by a too-single form of data enhancement. Insertion, replacement, and reordering do not allow the model to reach the local information of the sequence. When combined with masking or cropping operations, the model achieves good results on the data. It can be found that the best results can already be achieved using only the insertion and replacement operations. This may be because these two operations are more helpful in utilizing the association information. In addition, it can be found that the reordering operation helps to improve the recommendation effect. This indicates that reordering, although it destroys the order relations in the sequence, helps improve the model's generalization in the comparison learning task.

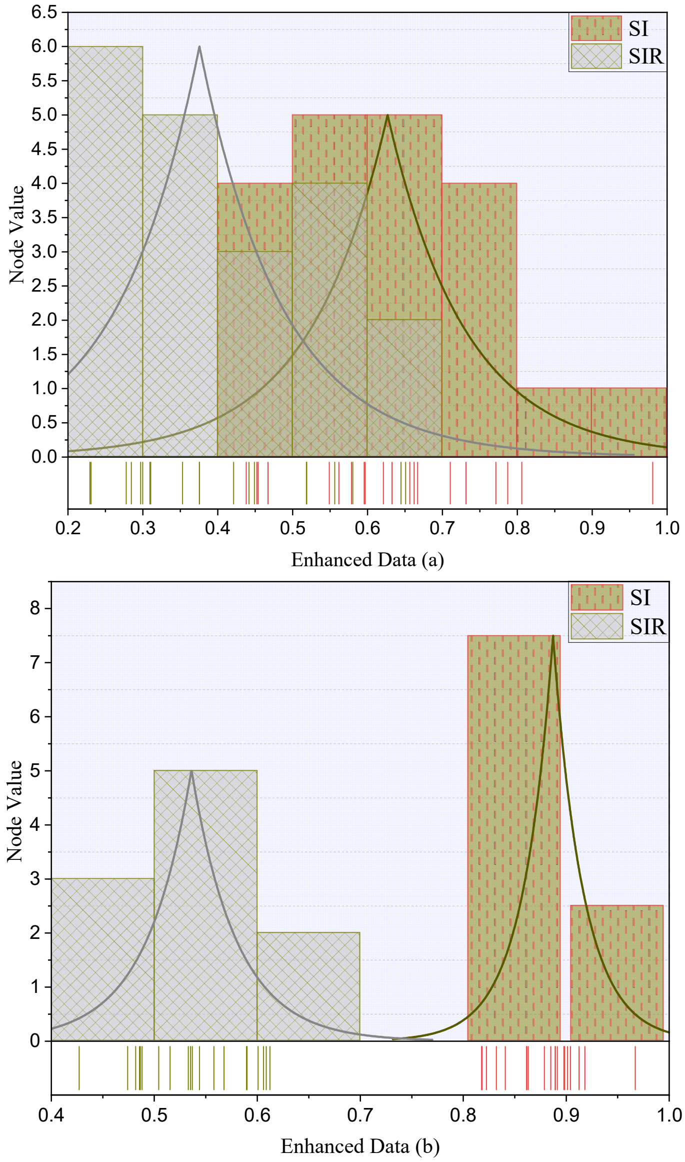

Figure 8. Impact of network-enhanced knowledge mapping on case analysis.

According to the analysis, the GCN+GAP and the GCN+GMP model achieved comparable accuracy in effect, indicating that each node in a case has a particular influence on the nature of the whole point. The critical nodes can clearly distinguish a chance, while the combined information of each node can also effectively distinguish an issue and calculate the similarity between different cases. A comprehensive analysis can conclude that in the vertical field of legal case similarity calculation, the use of GCN can effectively enrich the characteristics of case entities by exploiting the critical role of legal relationships in a case. Combined with the Bi LSTM network, the information of each node in a case carved by graph structure can be effectively aggregated to obtain a graph embedding framework reflecting the nature of the case.

5. Conclusions

This paper focuses on critical technology research about knowledge graph construction in security cases. The knowledge graph is composed of entities and relationships; therefore, this paper focuses on how to discover entities and mine the affinities between them. In terms of naming entities, in response to the problem that the traditional model cannot reflect the central word and cannot effectively distinguish the role of forward and reverse text, we propose to strengthen the main dish and the reverse sequence model, which can have a particular enhancement effect on entity recognition. This paper focuses on the problem of calculating the similarity of legal cases based on knowledge graph and deep graph neural network, which has a wide range of applications in legal work. First, this paper briefly reviews the development of text similarity computation at the present stage. It introduces the background knowledge related to the knowledge graph, GCN, and Bi LSTM. Secondly, this paper embeds the case knowledge map. It constructs a similarity computation network by combining GCN and Bi LSTM, which can effectively capture the case's long-term, global, and discontinuous dependencies among legal elements. Additionally, the proportion of negative sample mining is dynamically adjusted during the training process; it can effectively improve the model effect in the similarity learning problem and alleviate the shortage of legally labeled data. Through experiments, it is demonstrated that the method proposed in this paper achieves a high accuracy rate in case similarity calculation.

With the continuous development and progress of computer technology, all walks of life are speeding up the construction of information technology. As one of the essential functions in society, the public security department is also continuously constructing information technology. In the future development of case information management systems, it is worthwhile to improve by adding multimedia information, such as pictures, sound, and video information, to make case information more visualized and to design a more flexible and faster way to enter case information. This paper is based on the actual needs of the graph neural network enhanced knowledge graph framework for intelligent analysis of security cases. This work can also meet the specific needs of intelligent analysis of security case information management business. In the future, we will work to design and code the implementation of the philosophical analysis of the security case information management system based on the research results of this paper.

Conflict of interest

The author declares there is no conflict of interest.

References

1. Y. Li, H. Ma, L. Wang, S. Mao, G. Wang, Optimized content caching and user association for edge computing in densely deployed heterogeneous networks, *IEEE Trans. Mob. Comput.*, **21** (2022), 2130−2142. https://doi.org/10.1109/TMC.2020.3033563

2. Z. Guo, K. Yu, A. K. Bashir, D. Zhang, Y. D. Al-Otaibi, M. Guizani, Deep information fusion-driven POI scheduling for mobile social networks, *IEEE Network*, **36** (2022), 210−216. https://doi.org/10.1109/MNET.102.2100394
3. S. Xia, Z. Yao, Y. Li, S. Mao, Online distributed offloading and computing resource management with energy harvesting for heterogeneous MEC-enabled IoT, *IEEE Trans. Wireless Commun.*, **20** (2021), 6743−6757. https://doi.org/10.1109/TWC.2021.3076201
4. Q. Zhang, Z. Guo, Y. Zhu, P. Vijayakumar, A. Castiglione, B. B. Gupta, A deep learning-based fast fake news detection model for cyber-physical social services, *Pattern Recognit. Lett.*, **168** (2023), 31−38, https://doi.org/10.1016/j.patrec.2023.02.026
5. Z. Guo, K. Yu, A. Jolfaei, F. Ding, N. Zhang, Fuz-Spam: label smoothing-based fuzzy detection of spammers in Internet of Things, *IEEE Trans. Fuzzy Syst.*, **30** (2021), 4543−4554. https://doi.org/10.1109/TFUZZ.2021.3130311
6. L. Zhao, Z. Yin, K. Yu, X. Tang, L. Xu, Z. Guo, et al., A fuzzy logic based intelligent multi-attribute routing scheme for two-layered SDVNs, *IEEE Trans. Netw. Serv. Manage.*, **19** (2022), 4189−4200. https://doi.org/10.1109/TNSM.2022.3202741
7. Z. Cai, X. Zheng, J. Yu, A differential-private framework for urban traffic flows estimation via taxi companies, *IEEE Trans. Ind. Inf.*, **15** (2019), 6492−6499. https://doi.org/10.1109/TII.2019.2911697
8. L. Huang, R. Nan, K. Chi, Q. Hua, K. Yu, N. Kumar, et al., Throughput guarantees for multi-cell wireless powered communication networks with non-orthogonal multiple access, *IEEE Trans. Veh. Technol.*, **71** (2022), 12104−12116. https://doi.org/10.1109/TVT.2022.3189699
9. X. Shen, G. Shi, H. Ren, W. Zhang, Biomimetic vision for zoom object detection based on improved vertical grid number YOLO algorithm, *Front. Bioeng. Biotechnol.*, **10** (2022), 905583. https://doi.org/10.3389/fbioe.2022.905583
10. G. Shi, X. Shen, L. Gu, S. Weng, Y. He, Multipath interference analysis for low power RFID sensor under metal medium environment, *IEEE Sens. J.*, **19** (2023) 279−287. https://doi.org/10.1109/JSEN.2023.3253571
11. D. Peng, D. He, Y. Li, Z. Wang, Integrating terrestrial and satellite multibeam systems toward 6G: techniques and challenges for interference mitigation, *IEEE Wireless Commun.*, **29** (2022), 24−31. https://doi.org/10.1109/MWC.002.00293
12. C. Zhang, G. Zhou, F. Chang, X. Yang, Learning domain ontologies from engineering documents for manufacturing knowledge reuse by a biologically inspired approach, *Int. J. Adv. Manuf. Technol.*, **106** (2020), 2535−2551. https://doi.org/10.1007/s00170-019-04772-1
13. Z. Guo, K. Yu, Z. Lv, K. K. R. Choo, P. Shi, J. J. P. C. Rodrigues, Deep federated learning enhanced secure POI microservices for cyber-physical systems, *IEEE Wireless Commun.*, **29** (2022), 22−29. https://doi.org/10.1109/MWC.002.2100272
14. X. Zheng, Z. Cai, Privacy-preserved data sharing towards multiple parties in industrial IoTs, *IEEE J. Sel. Areas Commun.*, **38** (2020), 968−979. https://doi.org/10.1109/JSAC.2020.2980802
15. C. Zhang, G. Zhou, J. Li, F. Chang, K. Ding, D. Ma, A multi-access edge computing enabled framework for the construction of a knowledge-sharing intelligent machine tool swarm in Industry 4.0, *J. Manuf. Syst.*, **66** (2023), 56−70. https://doi.org/10.1016/j.jmsy.2022.11.015

16. T. Guo, X. Lu, K. Yu, Y. Zhang, W. Wei, Integration of light curve brightness information and layered discriminative constrained energy minimization for automatic binary asteroid detection, *IEEE Trans. Aerosp. Electron. Syst.*, **58** (2022), 4984−4999. https://doi.org/10.1109/TAES.2022.3192804
17. Y. Lu, L. Yang, S. X. Yang, Q. Hua, A. K. Sangaiah, T. Guo, et al., An intelligent deterministic scheduling method for ultra-low latency communication in edge enabled industrial Internet of Things, *IEEE Trans. Ind. Inf.*, **19** (2023), 1756−1767. https://doi.org/10.1109/TII.2022.3186891
18. C. Jin, D. Chen, Z. Lin, Z. Liu, M. Wu, How do you visit: identifying addicts from large-scale transit records via scenario deep embedding, *GeoInformatica*, **25** (2021), 799−820. https://doi.org/10.1007/s10707-021-00448-9
19. H. Yang, R. Ke, Z. Cui, Y. Wang, K. Murthy, Toward a real-time Smart Parking Data Management and Prediction (SPDMP) system by attributes representation learning, *Int. J. Intell. Syst.*, **37** (2022), 4437−4470. https://doi.org/10.1002/int.22725
20. F. Firouzi, B. Farahani, M. Daneshmand, K. Grise, J. Song, R. Saracco, et al., Harnessing the power of smart and connected health to tackle COVID-19: IoT, AI, robotics, and blockchain for a better world, *IEEE Internet Things J.*, **8** (2021), 12826−12846. https://doi.org/10.1109/JIOT.2021.3073904
21. F. Piccialli, V. S. Di Cola, F. Giampaolo, S. Cuomo, The role of artificial intelligence in fighting the COVID-19 pandemic, *Inf. Syst. Front.*, **23** (2021), 1467−1497. https://doi.org/10.1007/s10796-021-10131-x
22. A. Chakraborty, D. Mukherjee, S. Mitra, Development of pedestrian crash prediction model for a developing country using artificial neural network, *Int. J. Inj. Control Saf. Promot.*, **26** (2019), 283−293. https://doi.org/10.1080/17457300.2019.1627463
23. A. D'Alconz, I. Drago, A. Morichetta, M. Mellia, P. Casas, A survey on big data for network traffic monitoring and analysis, *IEEE Trans. Netw. Serv. Manage.*, **16** (2019), 800−813. https://doi.org/10.1109/TNSM.2019.2933358
24. G. Shi, X. Shen, F. Xiao, Y. He, DANTD: a deep abnormal network traffic detection model for security of industrial Internet of Things using high-order features, *IEEE Internet Things J.*, **20** (2023), 1−10. https://doi.org/10.1109/JIOT.2023.3253777
25. V. Hassija, V. Saxena, V. Chamola, F. R. Yu, A parking slot allocation framework based on virtual voting and adaptive pricing algorithm, *IEEE Trans. Veh. Technol.*, **69** (2020), 5945−5957. https://doi.org/10.1109/TVT.2020.2979637
26. K. H. de Jesus Prado, L. S. Souza, I. D. de Jesus Junior, M. C. Júnior, Applied intelligent data analysis to government data related to criminal incident: a systematic review, *J. Appl. Secur. Res.*, **15** (2020), 297−331. https://doi.org/10.1080/19361610.2020.1716511
27. C. Jin, G. Zhang, M. Wu, S. Zhou, T. Fu, Textual content prediction via fuzzy attention neural network model without predefined knowledge, *China Commun.*, **17** (2020), 211−222. https://doi.org/10.23919/JCC.2020.06.017
28. P. M. Asaro, AI ethics in predictive policing: from models of threat to an ethics of care, *IEEE Technol. Soc. Mag.*, **38** (2019), 40−53. https://doi.org/10.1109/MTS.2019.2915154
29. M. A. AlGhamdi, M. A. Khan, Intelligent analysis of Arabic tweets for detection of suspicious messages, *Arabian J. Sci. Eng.*, **45** (2020), 6021−6032. https://doi.org/10.1007/s13369-020-04447-0

30. X. Ferrer, T. van Nuenen, J. M. Such, M. Coté, N. Criado, Bias and discrimination in AI: a cross-disciplinary perspective, *IEEE Technol. Soc. Mag.*, **40** (2021), 72−80. https://doi.org/10.1109/MTS.2021.3056293
31. A. Zyner, S. Worrall, E. Nebot, Naturalistic driver intention and path prediction using recurrent neural networks, *IEEE Trans. Intell. Transp. Syst.*, **21** (2019), 1584−1594. https://doi.org/10.1109/TITS.2019.2913166
32. S. Asadianfam, M. Shams, A. R. Kenari, TVD-MRDL: traffic violation detection system using MapReduce-based deep learning for large-scale data, *Multimedia Tools Appl.*, **80** (2021), 2489−2516. https://doi.org/10.1007/s11042-020-09714-8
33. S. Raaijmakers, Artificial intelligence for law enforcement: challenges and opportunities, *IEEE Secur. Privacy*, **17** (2019), 74−77. https://doi.org/10.1109/MSEC.2019.2925649
34. A. Lavorgna, S. E. Middleton, B. Pickering, G. Neumann, FloraGuard: tackling the online illegal trade in endangered plants through a cross-disciplinary ICT-enabled methodology, *J. Contemp. Crim. Justice*, **36** (2020), 428−450. https://doi.org/10.1177/1043986220910297
35. D. Carneiro, A. Amaral, M. Carvalho, L. Barreto, An anthropocentric and enhanced predictive approach to smart city management, *Smart Cities*, **4** (2021), 1366−1390. https://doi.org/10.3390/smartcities4040072