ISSN 2616-5775 Vol. 7, Issue 1: 78-85, DOI: 10.25236/AJCIS.2024.070112

Entity Recognition Method for Key Information of Police Records Based on Bert-Bilstm-Selfatt-Crf

Xiaoang Chen[1,a,*], Xin Tong[1,b], Hailun Lin[2,c], Yunfei Xing[1,d]

[1]*School of Information Network Security, People's Public Security University of China, Beijing, China*
[2]*Institute of Information Engineering, CAS, Beijing, China*
[a]*yunchouweibo@qq.com*, [b]*tongxindotnet@outlook.com*, [c]*linhailun@iie.ac.cn*, [d]*406540085@qq.com*
[*]*Corresponding author*

Abstract: *At present, the public security department has a large number of case records, which contain information such as the alarm people, the time of the crime, the place of the crime, the modus operandi, the carrier involved, and the amount of money involved. Maximizing the use of this information is one of the key factors for the police to quickly solve the case and prevent crime. Using information extraction[1] technology to automatically classify police information and extract entities from case records can not only improve the efficiency of information analysis and enable the police to clarify the context of the case more quickly, but also have more significance for the further use of these information.*

Keywords: *Deep learning, Pre training language model, Alarm entity identification, Bert, Bilstm, CRF*

1. Introduction

With the continuous advancement of the strategy of reinvigorating the police through science and technology, massive unstructured police situation text data has been generated. The four categories of cases with the highest proportion in a certain city in one year are telecommunications network fraud cases, theft cases, intentional injury cases, and extortion cases. Statistics show that there are nearly 4,000 cases of the above four types in the city in two years, which contain valuable information about the involved elements such as the reporting person, reporting time, reporting location, criminal modus operandi, involved tools, and involved amount[2]. Tools are needed to help public security police organize massive data for case analysis and reasoning, so as to accelerate the cracking of cases and prevent crimes. If manual retrieval is carried out on these involved police texts, it will consume a lot of time and energy from public security personnel, and machine learning cannot fully understand the deep meaning of unstructured involved texts. Therefore, it is considered to use the information extraction technology of deep learning in natural language processing technology[3][4] to convert the involved text data into structured information, which can not only help the public security business personnel improve work efficiency and reduce labor costs, but also help to provide the query ability of the involved text and facilitate the research of downstream judicial applications, which is of great significance to the public security business personnel and researchers in the field of public security.

2. Data Set Construction

Currently, there is little research on the application of NER[5] technology in real-life police operations. Due to the confidentiality and sensitivity of alarm records, police situation texts must undergo desensitization processing or be annotated on the internal network of the police, and cannot be publicly available on the Internet, resulting in a lack of publicly available police situation entity extraction data sets. Therefore, constructing a corresponding police situation data set for the task of police situation entity extraction has become the premise and foundation of research on intelligent analysis of police situation. In order to conduct research on specific police situations in the city, more than 4,000 real case records from the city's police system were collected. All data were desensitized. 3,258 sentences and 74,590 words were selected to construct a police situation data set required for training and verification of the police situation text classification and entity extraction model. At the same time, combined with domain expert knowledge and research and summarization of relevant literature in the field of police intelligence, during the entire data preprocessing process, entities related

ISSN 2616-5775 Vol. 7, Issue 1: 78-85, DOI: 10.25236/AJCIS.2024.070112

to the case and similar words of the entity are extracted from the text. Eventually, a domain dictionary containing 3,892 named entities of case text is constructed.

Table 1: Definition table of four types of common police record entities in a city

Entity type	Entity description
A person who reports a crime	The police person in the police record of the case
Place of occurrence	The place where the case occurred in the police incident record
Time.	The date of the police report in the police case record
Action.	The behavior of misleading the alarm person to be deceived in the telecommunications network fraud alarm record
A carrier	The APP used by the alarm person to be deceived in the telecommunications network fraud alarm record
The target	The purpose of the alarm person in the telecommunications network fraud alarm record
Fraud amount	The amount involved in the telecommunications network fraud alarm record
Stolen amount	The amount involved in the theft police incident record
Stolen goods	Stolen items in the police report of theft
The victim	The victim in the intentional injury police record
Suspects	Suspect in the intentional injury police report
Injury condition	The victim's injuries in the intentional injury police incident record
Extortion amount	The amount involved in the police record of extortion and blackmail

2.1. Entity Definition

Before performing the task of entity extraction, it is necessary to first define the entities in the data. By summarizing and analyzing the descriptive characteristics of a large number of police incident texts, and counting the types of incidents that rank highly in a certain city, 13 common entities in telecommunications network fraud cases, theft cases, intentional injury cases, and extortion cases are summarized, as shown in Table 1 above.

2.2. Entity Annotation Methods and Descriptions of the Police Records

In order to efficiently complete the data annotation task, this article uses the open source corpus annotation platform Doccano as the annotation tool. Doccano is an efficient text annotation tool that can be used to construct labeled datasets to support tasks such as entity extraction, relationship extraction, and awareness recognition. The entity sequence in this article uses the BIO annotation scheme, with 16 labels:

B-Amount, I-Amount.

B-injury, I-injury

B-victim, I-victim

B-carrier, I-carrier

B-victim, I-victim

B-suspect, I-suspect

B-the reporter, I-the reporter

B-action, I-action

B-time of reporting, I-time of reporting

B-target, I-target

B-time, I-time

ISSN 2616-5775 Vol. 7, Issue 1: 78-85, DOI: 10.25236/AJCIS.2024.070112

B-stolen amount, I-stolen amount

B-Fraud amount, I-Fraud amount

B-stolen goods, I-stolen goods

B-location, I-location

O

There are 6 entities in the telecommunications network fraud case, including the reporter, the time of reporting, the location of reporting, the involved carrier, the modus operandi, and the amount of fraud. There are 5 entities in the theft case, including the reporter, the time of reporting, the location of theft, the amount of theft, and the stolen items. There are 5 entities in the intentional injury case, including the time of reporting, the location of reporting, the victim, the suspect, and the injury. There are 5 entities in the extortion case, including the reporter, the suspect, the location, the time, and the extortion amount. The entity annotation of involved texts is to prepare data for named entity recognition (NER) and text classification research. This annotation process is based on 4000 pieces of police intelligence data to construct a data set containing entity annotation to support subsequent model training and analysis.

3. Text Analysis Model For Police Cases

3.1. Overall Process of text Analysis of Police Cases

The analysis of police intelligence mainly includes two parts: classification of police intelligence and entity extraction. The main process includes data labeling module, data preprocessing module, classification of police intelligence and entity extraction module. The overall information extraction process designed for police records involving cases is shown in Figure 1.

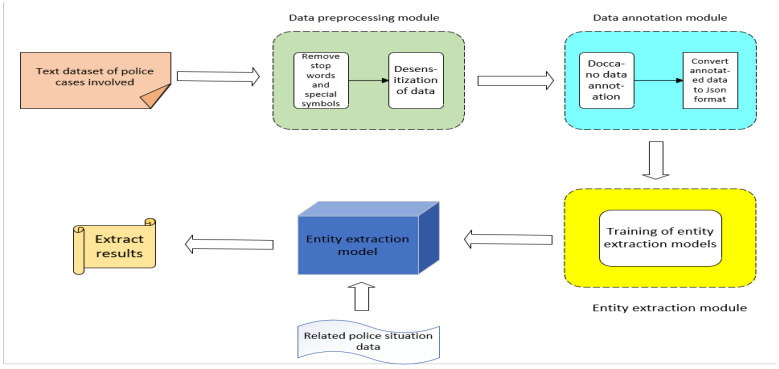

Figure 1: Overall flow chart of police records entity extraction

3.2. Classification of Police Situation Text Involved in Cases

3.2.1. Data Augmentation

Because the constructed police intelligence dataset involving the case is relatively small, the advantage of deep learning over machine learning algorithms is not obvious. The text data of police intelligence involving the case is relatively small, and after the division of the dataset, the sample data of the training set is even smaller, which leads to problems of poor generalization and overfitting when training the training set. Therefore, it is better to perform data augmentation on the constructed dataset. In the case of a small amount of data, using multiple data augmentation methods is better than using only one data augmentation method.

▪ Substitution entities: For example, "The alarm person Liu Zheng reported that he was defrauded of 50,000 yuan by using the Guangtai APP", you can replace "Guangtai" with "Wukong", that is, randomly select one from the APP_TYPE dictionary to replace the position, and other entities are also processed in this way.

▪ Randomly deleting characters: Randomly deleting or adding a character in Chinese without affecting people's understanding of the sentence. Considering that adding or deleting additional

ISSN 2616-5775 Vol. 7, Issue 1: 78-85, DOI: 10.25236/AJCIS.2024.070112

characters can have a certain impact on the semantics, this article only operates on 12% of sentences.

▪ Replace the position of adjacent Chinese characters. Randomly swap the positions of adjacent characters, with the principle of replacement being that it does not affect people's understanding of the semantic meaning of the sentence.

▪ Using translation software for back translation. By calling the API of Sogou Translation for batch translation, the back translation of the police situation text involved can be enhanced, that is, Chinese->English->Chinese.

3.2.2. BERT Alarm Classification Process

The BERT model is mainly used for two steps in the classification of police situation text. The first step is data augmentation. Due to the small amount of original data and the poor training effect of the model, the data is expanded based on the original data through the data augmentation method in the previous section, and the expanded dataset is used to train the model again. The second step is to classify the police situation text data using the BERT mode. The Bert model adopts the bert-base-chinese model open-sourced by Harbin Institute of Technology to add a linear classification layer for fine-tuning. According to the task of case classification, the police situation is divided into four categories: fraud, theft, intentional injury, and extortion. The supervised training is performed on the labeled dataset to output the feature vector of this article. Through the process of fine-tuning, the weights of the model are changed, making the model obtain better representation on this dataset and improve the classification effect. At the input end of the case classification task, in order to reflect the classification representation of this text, a special mark [CLS] is added at the beginning of the text sequence. The preprocessed police situation text involving cases is input into this model. The words in the text are vectorized through word embedding layer, including position embedding vector, word embedding vector, and sentence embedding vector. For example, "Bao" is the input representation of the fourth word in the sentence, which is obtained by adding position embedding E3, word embedding E and sentence embedding EA, as shown in Figure 2.

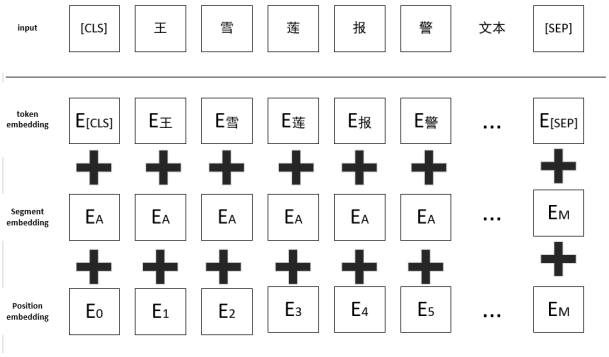

Figure 2: Schematic diagram of word vector

3.3. BERT-BiLSTM-SelfAtt-CRF Entity Recognition Model

This paper constructs the Bert bilstm selfatt CRF model for the police information text data set involved. The overall model structure is shown in Figure 3. The overall extraction model can be divided into four layers. First, each input Chinese character is represented by three embedded words. The pre training language model bert[6] generates the corresponding word vector based on the addition and embedding of three words in each character. Later, the bilstm layer allows the model to better capture contextual semantic information. Then, the self attention layer is used to enhance the capture of key information, so as to better obtain the long-distance dependence in the involved text. Finally, the CRF layer is used to realize the optimal sequence annotation of alarm text entity information.

ISSN 2616-5775 Vol. 7, Issue 1: 78-85, DOI: 10.25236/AJCIS.2024.070112

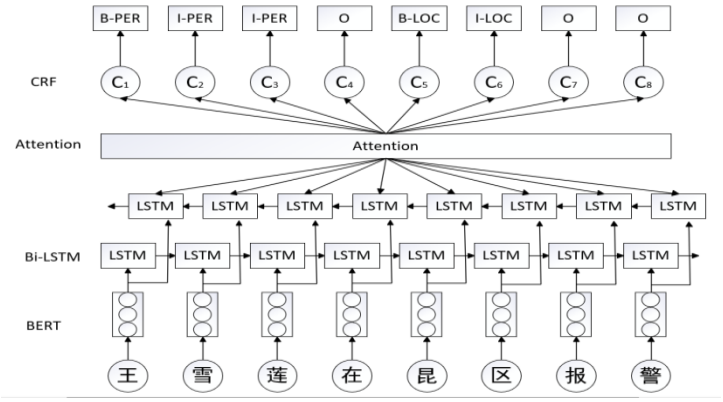

Figure 3: Overall model structure

BERT embedding layer. The word embedding layer of this article uses the BERT model. The input layer transfers the text data of the involved text to the BERT model, and the BERT model encodes the layer for word embedding. In the embedding layer, the BERT model maps the text of the involved text training set into a dynamic word vector representation, and passes the word vector to the BILSTM layer. Suppose that the input involved police situation text data is represented as $x=\{x_1,x_2,\ldots,x_t,\ldots,x_n\}$, where the t-th word of the sentence X can be represented by x_t. After the BERT model forms a word vector representation for the police situation text as e_t, where e^c is a word vector lookup table used during pre-training of the BERT model. The formula is shown in formula (1).

$$e_t = e^c(x_t) \tag{1}$$

BiLSTM layer. The model uses BILSTM[7] to extract text features. BILSTM can capture bidirectional text features when extracting text data, avoiding the loss of some text information due to long input text sequences. After the input alarm data is mapped into word vectors $e_t$by the BERT model, the vectorse$_t$ are input into the BiLSTM layer to extract the text features of the alarm data. The LSTM calculation steps are as follows.

Forgetting gate: Input is the hidden layer state h_{t-1} at the previous time step and the current input sequence x_t. Select the information that needs to be discarded. W_f and b_f are the weight matrix and bias term, respectively. σ is the activation function. The calculation formula is shown in formula(2):

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2}$$

Memory gate and current unit state: Select the information to be retained.

Output result and current hidden state: The bilstm extraction of bidirectional text feature representation formula is shown in equations (3), (4), and (5).

$$\overrightarrow{h_t} = \overrightarrow{LSTM}(e_t, \overrightarrow{h_{t-1}}) \tag{3}$$

$$\overleftarrow{h_t} = \overleftarrow{LSTM}(e_t, \overleftarrow{h_{t+1}}) \tag{4}$$

$$h_t = \overrightarrow{h_t} \oplus \overleftarrow{h_t} \tag{5}$$

In the formula, the forward propagation LSTM hidden state output at time t is $\overrightarrow{h_t}$, and the backward propagation LSTM hidden state output is $\overleftarrow{h_t}$. et is the input word vector at the current time. The hidden state output h_t of BiLSTM at time t is obtained by splicing the forward output $\overrightarrow{h_t}$ and backward output $\overleftarrow{h_t}$.

In order to solve the problem that RNN cannot effectively handle long-distance dependencies, LSTM was created. The forward LSTM network and backward LSTM network are stacked to produce a bidirectional long short-term memory network, which uses forward and backward LSTM networks to obtain historical information and future information respectively, thus obtaining more context-dependent information. The information about the past and the future is the two hidden states of the forward and backward LSTM networks. The vector generated by their splicing provides complete contextual history and future information, and the output of forward and backward is a fused result. All words or characters in the sentence are input to the network in the form of vectors. By using BiLSTN to encode Chinese, both forward semantic information and backward semantic information are treated equally, thus obtaining named entities in the text.

ISSN 2616-5775 Vol. 7, Issue 1: 78-85, DOI: 10.25236/AJCIS.2024.070112

Self-Attention layer. Although the BILSTM model has obvious effects in extracting text features, it cannot distinguish the importance of words when extracting features. For example, "alarm/person/yang/wei/alarm/reported/was/person/hit". This alarm record has 9 input words after word segmentation. The hidden state of each word after BILSTM extraction is the same. Therefore, this section introduces an attention mechanism[8] after BILSTM feature extraction to enable the model to autonomously distinguish important words. In the example above, "alarm reported" is an unimportant word, and all alarm information will have the word "alarm". This section uses attention mechanism to enhance the fusion of text output features from various nodes of BILSTM, so that key text features receive equal weights and improve the extraction ability of the model. The formula for calculating word weight using attention mechanism is shown in Equations (6) and (7).

$$A_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T^X} \exp(e_{jk})} \tag{6}$$

$$o = \sum_{i=1}^{n} a_i h_i \& \tag{7}$$

The formula A_{ij} is the attention weight, which is the degree of attention that element i has for element j. The larger the value of A_{ij}, the higher the degree of attention. The smaller the value of A_{ij}, the lower the degree of attention. E_{ij} is the influence of element j on current position i. T^X is the set of all elements, and h_i is the feature vector output by BILSTM. The main function of formula 4.5 is to automatically learn the attention weights between various elements and automatically capture the correlation of hidden layer information.

CRF layer. In the BiLSTM layer, the final hidden state of the BiLSTM network is spliced and calculated to obtain the score of each word belonging to each label. NER can be regarded as a sequence labeling problem. If there is no CRF layer, the label with the highest score in the BiLSTM layer can be directly selected. However, BiLSTM only considers context information in the alarm record, without considering the dependencies between labels, so it cannot guarantee that a meaningful label sequence can be output. CRF[9] is a discriminative undirected graph machine learning model that can add many constraints to ensure that the final prediction is valuable. The input of the CRF layer is the alarm record sequence x=(x1, x2, …, xt), and the output is the optimal label sequence y=(y1, y2, …, yt). First, formula 3.7 is used to calculate the position score of the label sequence. In this formula, P is the output matrix of the BiLSTM layer, and A is the transition score matrix, where A_{ij} represents the transition score from label i to label j.

$$score(x, y) = \sum_{i=1}^{m} P_{i,y_i} + \sum_{j=1}^{m+1} A_{y_{j-1},y_j} \tag{8}$$

As shown in Equation (9), the normalized probability is calculated using the Softmax function. In addition, for each training sample, the log-likelihood function is calculated using Equation (10).

$$P(y \mid x) = \frac{e^{score(x,y)}}{\sum_{y'=1}^{k} e^{score(x,y')}} \tag{9}$$

$$\log(y^x|x) = score(x, y^x) - \log \sum_y \exp\big(score(x, y')\big) \tag{10}$$

Finally, by maximizing the log-likelihood function and using the Viterbi algorithm in Equation (10), the label sequence with the highest score is selected as the prediction result.

$$y^* = \text{argmax}_{y'} score(x, y') \tag{11}$$

4. Experiment

4.1. Description of Experimental Environment and Training Parameters

In this chapter, the development language of the BERT-BiLSTM-SelfAtt-CRF model is Python 3.8. The model is implemented on the basis of the deep learning framework PyTorch 1.6.0. The hardware environment uses 14-core Intel Xeon E5-2680 v4 2.40Ghz, 16GB RAM and GPU:TITAN Xp (12GB), Windows 10 64-bit operating system.

The Chinese pre-training model with the BERT version of bert-base-chinese has a parameter count of 108M. The longest sequence length of the model is selected as 256, the learning rate is 6e-6, the batch size is 32, the dropout is 0.1, the epoch is 100, and the number of hidden units in the bilstm layer

ISSN 2616-5775 Vol. 7, Issue 1: 78-85, DOI: 10.25236/AJCIS.2024.070112

is 256. The loss rate used in the bilstm layer is 0.2. The training parameters of the model are shown in the table 2.

Table 2: Model training parameters

Bert base parametertable	
Maximum sequencelength	256
Training batchsize	64
Learningrate	6e-6
lossrate	0.1

4.2. Experimental Results and Analysis of Police Situation Classification

The police text classification dataset includes four types of text: telecommunications network fraud cases, theft cases, intentional injury cases, and extortion cases. 70% of the dataset is used as the training set, 20% as the validation set, and 10% as the test set.

The purpose of this experiment is to verify the impact of data augmentation methods on the performance of the BERT model in the task of police situation text classification. By introducing data augmentation, the model's performance is further improved, and various indicators are improved after data augmentation. To conduct this comparison, we selected the BERT model as a benchmark. Data augmentation is often used to improve the generalization ability of models on different tasks, especially in cases where there is a small amount of police situation text data. The comparative experimental results before and after data augmentation are shown in Table 3.

Table 3: Comparative experiment before and after data enhancement

model	periodization	Acc	P	R	F1
Bert	20	0.8622	0.8749	0.8637	0.8676
Bert+ data enhancement	20	0.9122	0.9216	0.9197	0.9208

4.3. Experimental Results and Analysis of Entity Extraction and Comparison of Police Records

In police combat, analysis, summarization, and judgment are extremely important tasks, and the accuracy of the model is an important indicator that cannot be ignored. Therefore, BERT-BILSTM-CRF is the most suitable baseline model for police situation entity extraction. Based on this, this paper introduces a multi-head self-attention mechanism to construct the BERT-BILSTM-SelfAtt-CRF model, which further improves the model's performance and outperforms other models in various metrics. The comparative experimental results of entity extraction for each model are shown in Table 4.

Table 4: Comparative experiment of alarm entity extraction

model	periodization	P	R	F1
CNN-LSTM	64	0.6249	0.5948	0.6219
BiLSTM-CRF	64	0.7136	0.6966	0.7051
BiGRU-CRF	64	0.6924	0.6823	0.6759
BiGRU-SelfAtt-CRF	64	0.7025	0.7278	0.7158
BERT-CNN-LSTM	64	0.7932	0.7665	0.7827
BERT-BiLSTM-CRF	64	0.8192	0.8159	0.8134
BERT-BiGRU-CRF	64	0.8059	0.7949	0.8047
BERT- BiLSTM -SelfAtt-CRF	64	0.8377	0.8169	0.8219

5. Conclusion

The multi head self attention mechanism in the Bert model and the bidirectional structure in the bilstm model ensure that the model can fully consider the context relationship in the alarm information and increase the accuracy of entity extraction. Self attention mechanism can ensure that the model can learn the internal structure of the text and capture the long-distance dependencies in the text. CRF layer can learn constraints from actual training data. At the tag level, the order between tags is considered to optimize the extraction effect. On the whole, the project can meet the needs of the actual work of public security and fill the gap of the current informatization of police work.

ISSN 2616-5775 Vol. 7, Issue 1: 78-85, DOI: 10.25236/AJCIS.2024.070112

Acknowledgment

This study is supported by the General Project for Research in Humanities and Social Sciences in Universities of Henan Province (No.2024-ZZJH-290).

References

[1] Liu Pengbo, Che Haiyan, Chen Wei. Overview of knowledge extraction technology [J]. computer application research, 2010 (9): 3222-3226
[2] Luo Dongmei, Liu Ruijun, Lin Xiping. Application of artificial intelligence language processing technology in unstructured case data [J]. Computer system applications, 2021,30 (04): 234-240
[3] DENGL, LIUY. Deep learning in natural language processing [M]. Berlin: Springer, 2018.
[4] He Y,Wu D,Beyazit E, et al. Supervised data synthesizing and evolving a framework for real-world trafficcrash severity classification [C] // IEEE. 2018 IEEE 30thInternational Conference on Tools with Artificial Intelligence (ICTAI).New York: IEEE,2018: 163-170.
[5] Nadeau D, Sekine S. A Survey of Named Entity Recognition and Classification [J]. Lingvisticae Investigationes, 2007, 30(1):3-26.
[6] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding [J]. arXiv preprint arXiv, 2018:1810.04805.
[7] SundermeyerM, SchlüterR, NeyH. Lstm Neural Networks for Language Modeling [C]. Thirteenth Annual Conference of the International Speech Communication Association, United States, 2012: 682-697.
[8] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Alignand Translate [J].Computer Science, 2014, 18(10):475-489.
[9] Charles A, et al. Conditional Random Fields: Factorized Probabilistic Models for Labeling and Segmenting Sequence Data [J]. Journal of Machine Learning Research, 2007, 8(14):642-661.