International Journal of Automation and Computing
www.ijac.net

18(6), December 2021, 935-946
DOI: 10.1007/s11633-021-1298-8

Fault Information Recognition for On-board Equipment of High-speed Railway Based on Multi-neural Network Collaboration

Lu-Jie Zhou[1] Jian-Wu Dang[1,2] Zhen-Hai Zhang[1]

[1] School of Automation and Electrical Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China

[2] Gansu Provincial Engineering Research Center for Artificial Intelligence and Graphic & Image Processing, Lanzhou 730070, China

Abstract: It is of great significance to guarantee the efficient statistics of high-speed railway on-board equipment fault information, which also improves the efficiency of fault analysis. Considering this background, this paper presents an empirical exploration of named entity recognition (NER) of on-board equipment fault information. Based on the historical fault records of on-board equipment, a fault information recognition model based on multi-neural network collaboration is proposed. First, considering Chinese recorded data characteristics, a method of constructing semantic features and additional features based on character granularity is proposed. Then, the two feature representations are concatenated and passed into the gated convolutional layer to extract the dependencies from multiple different subspaces and adjacent characters in parallel. Next, the local features are transmitted to the bidirectional long short-term memory (BiLSTM) to learn long-term dependency information. On top of BiLSTM, the sequential conditional random field (CRF) is used to jointly decode the optimized tag sequence of the whole sentence. The model is tested and compared with other representative baseline models. The results show that the proposed model not only considers the language characteristics of on-board fault records, but also has obvious advantages on the performance of fault information recognition.

Keywords: Train control system, Chinese named entity recognition (NER), character feature, gating mechanism, bidirectional long short-term memory (BiLSTM).

Citation: L. J. Zhou, J. W. Dang, Z. H. Zhang. Fault information recognition for on-board equipment of high-speed railway based on multi-neural network collaboration. *International Journal of Automation and Computing*, vol.18, no.6, pp.935–946, 2021. http://doi.org/10.1007/s11633-021-1298-8

1 Introduction

A high-speed railway is a complex modern engineering system. As the center of traffic safety control, the train control system plays an essential role in high-speed railway construction. The on-board equipment is the core technical equipment in the train control system, and it is the key factor to ensure the traffic safety and improve the transportation efficiency. When the on-board equipment of high-speed railway breaks down, the technical staff needs to record the fault information in the document one by one. The document is written in natural language and has the characteristics of extensive data scale. Due to the limitation of traditional unstructured data analysis technology, it is difficult to retrieve and analyze this kind of fault text data effectively. Therefore, it is very important to recognize valuable and uniformly formatted information from many unstructured natural language text data for intelligent statistics and association analysis. Fault information recognition for on-board equipment mainly extracts the valuable entity information from on-board fault records, including fault date, time, location, fault cause, fault analysis, treatment measure, etc. Therefore, this paper uses the named entity recognition (NER) technology to extract the specific domain information from the historical fault records, thus realizing the fault information recognition of on-board equipment.

According to the data format, data can be divided into structured data, semi-structured data, and unstructured data. Structured data refers to the data stored and organized by relational databases expressed by a two-dimensional table structure. Contrary to the former, unstructured data refers to information with no pre-defined data model or is not organized in a pre-defined manner. Unstructured information typically includes photos, video and audio files, text files, etc. Compared with the data stored in the database in the form of fields, the lack of structure makes unstructured data more challenging to

Research Article
Manuscript received October 7, 2020; accepted March 29, 2021; published online April 26, 2021
Recommended by Associate Editor Jie Zhang
Colored figures are available in the online version at https://link.springer.com/journal/11633

search, manage, and analyze. Semi-structured data is between structured data and unstructured data. This kind of data does not conform to relational databases such as structured query language (SQL) but contains some level of organization through semantic tags or metadata, such as hypertext markup language (HTML)[1]. The purpose of NER is to classify entities in unstructured text into pre-defined categories. NER has been widely used in the general field[2], as well as medical[3], military[4], chemical[5], and other specific fields. At present, the main technical methods of NER can be divided into rule-based methods, statistics-based methods, and deep learning-based methods. The rule-based method relies too much on the dictionary and rule base. This method has low recognition ability for ambiguous words and out-of-vocabulary words and poor cross-domain portability[6]. Statistics-based methods mainly include the hidden Markov model (HMM)[7], maximum entropy Markov model (MEMM)[8], and conditional random field (CRF)[9]. Among them, CRF has the best performance on usability, stability, and accuracy. CRF is a discriminative probability model, which optimizes the global parameters after considering all possible tag sequences and the correlation between adjacent tags. CRF overcomes the drawback of the independence hypothesis in HMM and solves the label bias problems in MEMM[10, 11]. However, all these methods are heavily relying on feature engineering and external resources. Such task knowledge is costly to develop, making sequence labeling models difficult to adapt to new tasks or new domains[12].

In recent years, recurrent neural network (RNN) and convolution neural network (CNN) are the two most commonly used deep learning models[13, 14]. In [13], the long short-term memory (LSTM) network with memory capability is used to deal with time series high correlation in the fault diagnosis of chemical processes. This paper optimizes the LSTM network based on the original LSTM neural network by adding the link to the traditional network to determine the optimal number of hidden layer nodes. In the leak detection of the water supply network, Hu et al.[14] divide the network into several leakage areas to reduce the number of categories based on spatial clustering. Each area′s number is marked as the category label of the multiscale fully convolutional networks (MFCN). Then, feature extraction and classification are realized by MFCN. In the current research, with the support of computational power and word distributed representation technology, the research focus of the NER task has gradually shifted to the deep learning field. Collobert et al.[15] proposed a model combining CNN and CRF to capture the depth features of sequential tagging tasks. It uses a simple feed-forward neural network to limit the use of context to a fixed-size window around each word, which discards useful long-distance relationships between words. The bidirectional LSTM-CRF models for sequence tagging was proposed in [16]. The model can use bidirectional LSTM to obtain past and future input characteristics. Thanks to the CRF layer, statement-level tag information can be used. However, their effectiveness is limited by the lack of high-quality word embedding and deep-seated features. Chiu and Nichols[17] proposed a hybrid network of LSTM and CNN to realize NER. This network can automatically detect English words and character-level features, eliminating the need for most feature engineering. Liu and Chen[18] proposed a Bi-CLSTM model for social media named entity relation extraction, and this model extracts relations via a hybrid model of LSTM and piecewise-CNN. The model takes into account the long-distance dependence of features and deep-seated feature extraction. Compared with traditional machine learning methods, deep learning has more advantages in feature learning. It can reduce the dependence on linguistic knowledge and complex feature engineering and has strong robustness and generalization ability.

Although the NER in the general field has achieved good results, most of them are related to English, few achievements have been made in Chinese. Moreover, there is still a lack of research in the high-speed railway field, which needs continuous research and expansion. At present, there are many challenges in the fault information recognition for on-board equipment of high-speed railway. Firstly, there is a lack of annotated corpus based on the on-board equipment field. Secondly, most of the descriptions in fault records are unstructured narrative information, which is not suitable to extract entity information only by grammatical structure. Third, the fault records contain a large number of named entities, which is a kind of knowledge-intensive text, and the density of entity distribution is higher than that of general domain text. Fourth, because the fault information is recorded by different technicians, the length of the entity and the expression of technical terms are various. For example, "重新启动(Restart)" can also be written as "重启(Restart)" when describing the treatment measure, "K2073+500" can also be written as "2 073.500 km" when describing the kilometer mark. Therefore, it is necessary to design input features and recognition models according to the actual on-board fault record language characteristics.

Based on the application requirements for on-board equipment fault information recognition, a recognition model considering multi-neural network collaboration is proposed in this paper. Some studies show that the word-based model is not as good as the character-based model in the research of Chinese named entity recognition based on machine learning[19, 20]. Therefore, the fault information descriptions are segmented with character granularity to avoid the error propagation caused by word segmentation errors. Considering the language characteristics of on-board fault records, a construction method of semantic feature representation is proposed. An additional feature representation based on character-level is pro-

posed to improve the adaptability of the entity recognition model to fault information. Then, semantic feature representation and additional feature representation are combined, and the gated convolutional layer is used to capture local context information. It can effectively capture local features and alleviate the disappearance of gradients in the training process. Next, feed the local context features into the BiLSTM to learn long-term dependency context information. On top of BiLSTM, the sequential CRF is used to jointly decode the optimized tag sequence of the whole sentence.

In order to verify the correctness and effectiveness of the model, this work compares the model with several other baseline models by using the corpus of the on-board equipment fault field. The experimental results show that the model has obvious advantages on the performance of fault information recognition.

2 Fault information recognition model of on-board equipment

The essence of the NER task is a kind of sequence tagging problem, so it is necessary to transform the on-board equipment fault information recognition task into a sequence tagging. Taking the character granularity as the basic word segmentation unit, each character is labeled by the BIOES (B: begin; I: inside; O: outside; E: end; S: single) method. According to the characteristics of fault description, two vector representation schemes of semantic features and additional features are used. The multi-neural network collaboration model based on BiLSTM-CRF combined with gated convolution is used as the fault information recognition model of on-board equipment. The structure of the model is shown in Fig. 1. It consists of four parts: an embedding layer, a gated convolutional layer, a BiLSTM layer, and a CRF layer. Each part of the model is described in detail in Sections 2.1–2.3.

2.1 Fault data tagging of on-board equipment

Each character in the corpus of the on-board equipment fault field is labeled using the BIOES method. "B-X" represents the beginning of the entity, "I-X" represents the interior of the entity, "E-X" represents the end of the entity, "S-X" represents a single entity, and "O" represents outside the entity. Previous studies have shown that since the BIOES method can obtain more abundant sequence position information, the labeling ac-

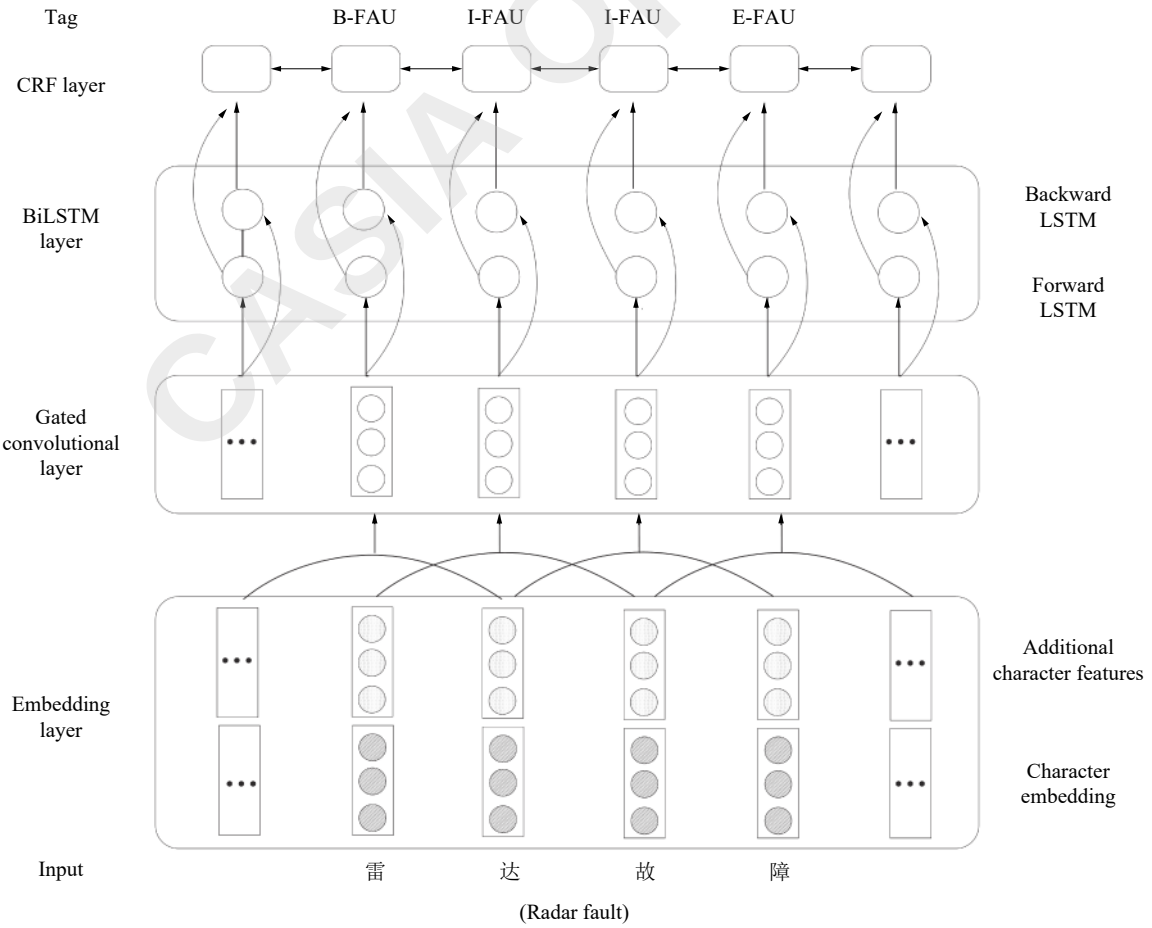

Fig. 1 Structure of multi-neural network collaboration model

curacy of the BIOES-based method is higher than that of the BIO-based method[21].

According to the on-board fault records contents, the fault named entity is divided into ten types: date, time, train number, vehicle number, terminal number, location, kilometer mark, fault cause, fault analysis, and treatment measure. To clearly explain the fault information recorded by the technical staff in the document, one of the records is taken as an example and labeled. Take "5月3日G2093次列车在藁城南—辛集南间K62+951处报测速雷达故障(Train G2093 reported speed radar failure at K62+951 between Gaocheng South and Xinji South on May 3)" as an example.

It can be seen from Table 1 that the fault records contain relatively dense entity information. Accurate recognition of the fault entity information can make sufficient preparation for intelligent retrieval, statistics, and association analysis.

2.2 Representation learning of fault data

2.2.1 Character-level distributed representation

The on-board fault information is recorded in the form of text, which needs to be transformed into a vector form. In traditional machine learning methods, the bag of word (BOW) model is usually employed to represent the vector features of words, such as the one-hot representation[22]. This model has some advantages in dealing with discrete data, but it ignores the order of words and the semantic relationship. In the past few years, non-linear neural networks with input distributed word representations, known as word embeddings, have been broadly applied to natural language processing problems with great success. This distributed representation can convert each word in the statements into a low-dimensional real-value vector. The word embedding of the current word is calculated using the text context representation through language models such as N-gram[23] and neural network[24].

Table 1 Example of character tags

Sentence	Tag	Sentence	Tag	Sentence	Tag
5	B-DAT	在	O	+	I-KIL
月	I-DAT	藁	B-LOC	9	I-KIL
3	I-DAT	城	I-LOC	5	I-KIL
日	E-DAT	南	I-LOC	1	E-KIL
G	B-TRA	–	I-LOC	处	O
2	I-TRA	辛	I-LOC	报	O
0	I-TRA	集	I-LOC	测	B-FAU
9	I-TRA	南	E-LOC	速	I-FAU
3	E-TRA	间	O	雷	I-FAU
次	O	K	B-KIL	达	I-FAU
列	O	6	I-KIL	故	I-FAU
车	O	2	I-KIL	障	E-FAU

When calculating the embedding of Chinese, the sentences are usually segmented with word granularity. However, the on-board fault records contain a large number of professional terms related to the railway field. If word segmentation is carried out, different word segmentation methods may get distinct recognition boundaries, resulting in entirely different sequence tagging results. The subsequent recognition model cannot judge whether the word segmentation is correct, which leads to an entity recognition error. In order to limit the influence of segmentation error propagation, fault information descriptions are segmented with character granularity. Word2vec[25] is a commonly used tool for learning embedding. Continuous BOW model word2vec model based on hierarchical softmax is used to represent characters in semantic feature vectors. After preprocessing, the fault records are represented as the serialized data. The character embedding matrix of the whole corpus $E \in \mathbf{R}^{m\times|V|}$ can be obtained, where m is the dimension of the character embedding, $|V|$ is the size of characters in the corpus. In the distributed vector representation based on character granularity, for a sentence $s = \{w_1, w_2, \cdots, w_n\}$, each character in the sentence will be mapped to an m-dimensional vector, that is $e_i \in \mathbf{R}^m$.

2.2.2 Additional character-level features

The entity information recognition of on-board fault records belongs to the category of natural language processing. The technical terms and specific descriptions are still in line with the railway specifications. Through some unique keywords and features, entity boundaries can be distinguished effectively.

In the fault records, numbers, upper case letters, and lower case letters usually exist in the entity of fault date, time, vehicle number, train number, terminal number, and kilometer mark. This information rarely appears in the non-entity content. Therefore, it is necessary to give additional feature labels to numbers and upper and lower case letters. Punctuation is often included in the entity of time, vehicle number, location, and kilometer mark. Additional feature labels also need to be given to the punctuation in the text. The entity name of the fault location is usually recorded in Chinese place-name according to the railway specifications. Location entity names usually include place-name elements such as "站(station)" and "南(south)". Tagging the place-name elements is helpful for the model to distinguish the boundaries of location entities. Different additional feature tags are mapped to a multi-dimensional continuous value vector $tag_i \in \mathbf{R}^k$ by vectorization, where tag_i is the additional feature vector of the i-th character, and k is the vector dimension. Additional features and tags are shown in Table 2.

2.3 Fault information recognition based on multi-neural network collaboration

To recognize the entity name of the on-board fault ef-

Table 2 Example of additional features and tags

Tag	Name	Example
Num	Number	G2093
Low	Lower case letter	62.950 km
Upp	Upper case letter	CRH380B-5 648
Pun	Punctuation (，/：/+/./−)	K1285+600
Pne	Place-name element (站(station)/所(block)/场(yard)/东(east)/南(south)/西(west)/北(north))	鸿宝线路所 (Hongbao block post), 徐兰场 (Xulan yard), 北京南 (Beijing south)
Oth	other	/

fectively, a multi-neural network collaboration model based on BiLSTM-CRF combined with gated convolution called GC-BiLSTM-CRF is proposed, which is illustrated in Fig. 1. The embedding layer converts input characters into embedding according to distributed representation and converts characters into feature vectors according to additional tags. These vectors are concatenated and fed to the gated convolutional layer for each character to extract the global dependencies from different multiple subspaces and arbitrary adjacent characters. The formed features are then fed into a bidirectional LSTM to extract context features and generate a feature matrix. On the basis of BiLSTM, a sequential CRF is used to jointly decode the tags for the whole sentence.

2.3.1 Gated convolutional layer

Convolutional neural networks are widely used in NER tasks to extract local information of text features. Pooling operation is used in most convolutional neural network models[17]. However, it is sensitive to pooling operations when extracting short text features. Li et al.[26] and Tang et al.[27] have verified through experiments that the pooling operation will cause information loss of the local position and sequence structure in the process of sequence modeling. In order to avoid the destruction of text sequence information by down-sampling, this paper uses a gated convolutional layer to capture local correlation information between contexts in parallel on the sequence[28].

The sentence is convoluted based on the character's granularity. Let $e_i \in \mathbf{R}^m$ be the m-dimensional vector corresponding to the i-th character in the sentence. The character-level distributed representation constructed by a sentence of length n can be expressed as

$$e_{1:n} = [e_1, e_2, \cdots, e_n]^{\mathrm{T}} \tag{1}$$

where $e_{1:n} \in \mathbf{R}^{n\times m}$, T is the transpose operation. Let $tag_i \in \mathbf{R}^k$ be the additional feature representation corresponding to the i-th character in the sentence. An additional feature matrix of a sentence with length n is expressed as follows:

$$tag_{1:n} = [tag_1, tag_2, \cdots, tag_n]^{\mathrm{T}} \tag{2}$$

where $tag_{1:n} \in \mathbf{R}^{n\times k}$. For each sentence, these vector representations are concatenated into a matrix $x \in \mathbf{R}^{n\times(m+k)}$ and fed to the gated convolutional layer.

$$x = e_{1:n} \oplus tag_{1:n} \tag{3}$$

where $\oplus$ is the concatenation operation. In this experiment, the length threshold of the sentences is set to maxlen, and the sentences whose length is less than maxlen are supplemented with 0.

Dauphin et al.[29] proved that reasonable use of the gating mechanism can effectively enhance the effect of natural language processing. The gated convolutional layer incorporates the gated unit into the traditional convolutional layer. Gated units control the path through which information flows in the network. Compared with traditional convolution, gated convolution retains the ability of non-linear operation and filters useless information. Compared with LSTM, gated convolution is a parallel hierarchical structure that can better capture abstract hierarchical features. Compared with the chain structure, the hierarchical structure reduces the number of non-linearities of a given context size, thus alleviating the vanishing gradient problem and simplifying learning[30].

For matrix $x \in \mathbf{R}^{n\times(m+k)}$, the gated convolutional layer output can be expressed as

$$m_i = (x_{i:i+h-1} * w_i + b_i) \otimes \sigma(x_{i:i+h-1} * v_i + c_i) \tag{4}$$

where $w_i \in \mathbf{R}^{h\times(m+k)\times l}$, $b_i \in \mathbf{R}^l$, $v_i \in \mathbf{R}^{h\times(m+k)\times l}$, $c_i \in \mathbf{R}^l$ are learned parameters, h is the filter window size, l is the number of filter windows, $m+k$ is the dimension of the input vector, $*$ denotes convolution operator, σ is the sigmoid function, and $\otimes$ is the element-wise product between matrices.

For L filters, the feature matrix can be obtained:

$$M = [m_1, m_2, \cdots, m_L]. \tag{5}$$

2.3.2 BiLSTM layer

LSTMs are variants of RNNs, which can alleviate the problem of gradient disappearance by incorporating past information[31, 32]. The basic structure of the LSTM unit is shown in Fig. 2. The LSTM unit consists of three multiplication gates that control the proportion of information forgotten and passed to the next time step. These gates are helpful to learn the long-distance dependence between contexts and solve the problem of association between word orders. Therefore, LSTM is used to learn the dependencies between text sequences.

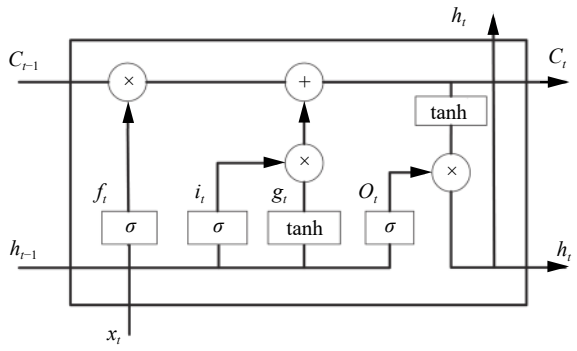

Fig. 2 Architecture of LSTM layer

Formally, the formulas to update an LSTM unit at time t are

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{6}$$

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{7}$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{8}$$

$$g_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \tag{9}$$

$$c_t = f_t \times c_{t-1} + i_t \times g_t \tag{10}$$

$$h_t = o_t \times \tanh(c_t) \tag{11}$$

where σ is the element-wise sigmoid function and $\times$ is the element-wise product. x_t is the input vector at time t and h_t is the hidden state vector storing all the useful information at time t. i_t, f_t and o_t represent the outputs of input gate, forget gate, and output gate at time t, respectively. c_t is the state vector at time t. W_i, W_f, W_o and W_c are the weight matrices for the hidden state h_t. b_i, b_f, b_o and b_c denote the bias vectors.

A single LSTM only considers the past context information, which is not affected by future context information. However, for the named entity recognition task, the past and future information impacts its recognition accuracy. Therefore, a bidirectional LSTM layer is used in this model. The principle is to use the forward and backward hidden states to extract the past and future information and then connect the two hidden states in series as the final output.

The matrix $M = [m_1, m_2, \cdots, m_L]$ obtained by the gated convolutional layer is fed into the BiLSTM layer. The output sequence for forward LSTM is $F = [f_1, f_2, \cdots, f_L]$, where the input of f_1 is m_1, and starting from f_2, the input of $\forall f_i \in F$ is $m_i \oplus f_{i-1}$. Similarly, the output sequence for backward LSTM is $B = [b_1, b_2, \cdots, b_L]$, where the input of b_1 is m_1, and starting from b_2, the input of $\forall b_i \in B$ is $m_i \oplus f_{i-1}$. Then, m_k can be calculated by the BiLSTM neural network to get the final output matrix p_k, and $p_k = f_k + b_k$. Finally, the matrix generated by BiLSTM layer is $P \in \mathbf{R}^{L \times T}$, where L is the number of features and T is the category of tags.

2.3.3 CRF layer

Named entity recognition can be regarded as a sequence labeling task. A common method is to add a softmax layer on top of the BiLSTM layer to decode the probability of each tag category[17]. The sequence tagging task can be completed by outputting the tag with the highest probability. Although this model uses BiLSTM to learn the dependencies between contexts, the tags decoded by the softmax layer are independent of each other. The softmax layer only outputs the tags with the highest probability value based on the current time and cannot learn the constraint relationship between the tags, resulting in an output of invalid tag sequences. In the named entity recognition task, the entity tag of a word is affected by both the word context and the word context tag. There are strong dependencies between tags. For example, the tag "B-X" cannot be followed by another tag "B-X".

Therefore, in the proposed model, the CRF layer is used to consider the dependencies between tags in neighborhoods rather than using the output of a softmax layer to make independent tag decisions. The CRF layer uses a state transition matrix as a parameter, which can effectively use past and future tags to predict current tags, obtain the global optimal tag sequence by the relationship between adjacent tags, and add constraints to the final predicted tags. The constraints involved include: 1) The first word in the sentence begins with the tag "B-" or "O-", and the tag starting with "O-" cannot be connected with the tag "I-" or "E-". 2) The tag "B-X1, I-X2, E-X3", X1, X2 and X3 should belong to the same entity. These constraints can reduce the probability of unreasonable sequences in tag sequence prediction.

Formally, $s = \{w_1, w_2, \cdots, w_n\}$ is used to represent a generic input sentence. The matrix $P \in \mathbf{R}^{L \times T}$ is the output of the BiLSTM layer, where $p_{i,j}$ is the probability value of row i and column j in the matrix. The transition matrix is defined as A, where $a_{i,j}$ represents the probability value of tag i transferring to tag j. The score function for generating prediction tags sequence $y = \{y_1, y_2, \cdots, y_n\}$ can be expressed as

$$score(s, y) = \sum_{k=1}^{n} p_{k,y_k} + \sum_{k=0}^{n+1} a_{y_k, y_{k+1}} \tag{12}$$

where p_{k,y_k} is obtained from the output matrix of BiLSTM and represents the probability that the k-th character in the sentence belongs to tag y_k. $a_{y_k, y_{k+1}}$ represents the probability of transfer from tag y_k to tag y_{k+1}. The softmax function is used to normalize $score(s, y)$, the probability of generating the prediction tag sequence y can be obtained as

$$p(y|s) = \frac{e^{score(s,y)}}{\sum_{\tilde{y} \in Y_s} e^{score(s,\tilde{y})}} \tag{13}$$

where e is Euler number, $\tilde{y}$ is the true tag, Y_s is all the sequence tags possible for the input.

During training, $p(y|s)$ needs to be maximized to obtain the optimal prediction tag sequence. It can be solved according to the maximum likelihood estimation, and the likelihood function of $p(y|s)$ can be obtained according to (14).

$$\log p(y|s) = score(s, y) - \log \sum_{\tilde{y} \in Y_s} e^{score(s,\tilde{y})}. \tag{14}$$

While decoding, the sequence with the highest output probability is used as the prediction sequence:

$$y* = \arg\max_{\tilde{y} \in Y_s} score(s, \tilde{y}). \tag{15}$$

3 Experiment

To verify the effectiveness of the proposed model GC-BiLSTM-CRF, the model is compared with the mainstream baseline model on the corpus of on-board equipment. The performance of the proposed model in the on-board information recognition task is evaluated from three aspects: 1) Discuss the influence of model parameters on experimental results. 2) Compare the proposed model with several strong baselines to evaluate the effectiveness of the model. 3) The effect of introducing additional features into the model is verified.

3.1 Dataset and evaluation metrics

Through the collation of the data, the on-board equipment corpus is obtained. The corpus information is shown in Table 3.

The data sets are divided into training sets, validation sets, and test sets according to the ratio of 8:1:1. And Python was used to implement the model through the Keras.

Generally, the performance of NER tasks can be estimated by calculating the correct entities identified by the model and the total named entities available in the corpus. In this research, the performance of the model was evaluated by precision (*Macro-P*), recall (*Macro-R*), and F1 score (*Macro-F*1), which can be computed by

$$Macro\text{-}P = \frac{1}{K} \sum_{i=1}^{K} P_i \tag{16}$$

$$Macro\text{-}R = \frac{1}{K} \sum_{i=1}^{K} R_i \tag{17}$$

where P_i and R_i represent the precision and recall of i. F1 score is a combination of recall and precision and helps to understand the results much better than the other metrics, as shown in (18). It is given by

$$Macro\text{-}F1 = \frac{1}{K} \sum_{i=1}^{K} F_i = \frac{1}{K} \sum_{i=1}^{K} \frac{2P_i R_i}{P_i + R_i}. \tag{18}$$

Table 3 Corpus entity statistics

Entity type	Number of entities	Number of characters
Date	411	4 286
Time	370	1846
Train number	398	1879
Vehicle number	348	2 829
Terminal number	321	614
Location	366	2 208
Kilometer mark	237	1977
Fault cause	736	5 413
Fault analysis	295	1967
Treatment measures	681	3 779

3.2 Parameter settings

In the experiment, the dimension of character embedding is set to 100. The model training is done by mini-batch with the Adam optimizer. Each mini-batch consists of multiple sentences with the same number of tokens. The optimal parameter values are determined by controlling a single variable during the experiments. In order to prevent the overfitting of the network, a dropout layer is added before the LSTM layer to make some connections drop out randomly. The dropout rate is set to 0.5. Based on this, the following experiments are carried out. The best recognition results are found by changing the filter window size, the number of filter windows, and the number of LSTM units in each layer. In order to find the appropriate parameters in the fixed ranges tested in this experiment, we set different filter window sizes such as 2, 3, 4 and 5 and the numbers of filter windows such as 50, 100, 150, 200, 250 and 300. Simultaneously, model is tuned with different numbers of LSTM units in each layer such as 50, 100, 150 and 200.

F1 score of various parameters are given in Tables 4–6. The models are trained using only the training set to isolate the effect of various parameters on both validation and test sets.

First, the number of filter windows is set to 100, and the number of LSTM units is set to 100 to verify the optimal value of filter window size. As shown in Table 4, the model proposed has good performance when the filter window size is 3. Then, the filter window size is set to 3, and the LSTM unit number remains unchanged to find the optimal number of filtering windows. As shown in Table 5, the performance is improved when the filter window number is 150. The filter window size is set to 3 and the number of filter windows is set to 150 to continue the experiment, as shown in Table 6. The experimental res-

Table 4 F1 score results with various filter window sizes

Filter window size	Validation	Test
2	0.897	0.867
3	**0.899**	**0.868**
4	0.883	0.861
5	0.894	0.859

Table 5 F1 score results with various filter window numbers

Filter window number	Validation	Test
50	0.870	0.864
100	0.899	0.868
150	**0.913**	**0.887**
200	0.889	0.867
250	0.910	0.883
300	0.892	0.861

Table 6 F1 score results with various LSTM unit numbers

LSTM units number	Validation	Test
50	0.851	0.840
100	**0.913**	**0.887**
150	0.912	0.882
200	0.907	0.871

ults show that the proposed model has good performance when the filter window size is 3, the number of filter windows is 150, and every layer has 100 LSTM units.

After determining the optimal values of the above parameters, the optimal dropout rate of the model is tested. All other parameters are the same as the best model obtained by the experiment. Dropout can reduce the risk of overfitting in model training by reducing the interaction between hidden layer nodes. The results of various dropout values are compared in Table 7, and it can be seen that the effect is best when the dropout rate is set to 0.5 at the beginning. Therefore, the parameters used in this experiment are shown in Table 8.

3.3 Model evaluation and comparison

Several representative models for the NER tasks are selected as the baseline, such as RNN, LSTM, CNN, and their variants. The proposed model is compared with other baseline models. The same dimension character embedding without additional features is used as input to the baseline models. Each model is tested with the optimal parameters to ensure the effectiveness of the comparative experimental results.

The results of all the NER models concerning the precision, recall and F1 score on the testset are shown in Table 9. It can be found that the performance of BiLSTM is better than that of LSTM and RNN neural networks

Table 7 F1 score results with various dropout values

Dropout	Validation	Test
0.2	0.911	0.874
0.3	0.907	**0.887**
0.4	0.908	0.876
0.5	**0.913**	**0.887**
0.6	0.896	0.871

Table 8 Hyper-parameter values

Parameters	Value
Character embedding dimension	100
Filter window size	3
Filter window number	150
LSTM units	100
Dropout	0.5
Optimizer	Adam
Epoch	100
Mini-batch size	9

when the character embedding is used as the input of the model. The F1 scores of RNN and LSTM are 0.614 and 0.677, respectively, while BiLSTM is 0.723, which is higher than that of the former two models. BiLSTM makes full use of past and future sequence information. Compared with RNN, it can alleviate the problem of gradient disappearance. The results show that the performance of LSTM is better than that of RNN.

Besides, it can be seen that the CRF layer promotes the performance of RNN and BiLSTM. The individual RNN produces 0.614 in the F1 score, while the RNN combined with CRF gives 0.729 in the F1 score. When BiLSTM is combined with the CRF layer, the F1 score increases to 0.824, while the F1 score for a single BiLSTM is only 0.723. It shows that CRF can make use of the relationship between adjacent tags to realize the global optimization of the tag sequence.

The performance of RNN and BiLSTM combined with CNN is also improved. The RNN with CNN improved from 0.614 to 0.730 in the F1 score. Compared with BiLSTM, the F1 score of CNN-BiLSTM is increased by 0.102. It shows the validity of the character-level features extracted by the convolutional layer.

This paper proposes a multi-neural network collaboration model based on BiLSTM-CRF combined with gated convolution. Compared with other baseline models, the proposed multi-neural network collaboration model achieves the highest precision, recall and F1 score. Moreover, compared with the second-ranked CNN-BiLSTM-CRF model, the proposed model increases 0.038 in the F1 score and 0.056 in the recall. It shows that the semantic features and additional features based on character granularity are concatenated and can improve the ad-

Table 9 Comparisons of the proposed model with baselines

Models	*Marco-P*	*Marco-R*	*Marco-F*1
RNN	0.696	0.549	0.614
CNN-RNN	0.788	0.679	0.730
RNN-CRF	0.797	0.673	0.729
CNN-RNN-CRF	0.769	0.758	0.764
LSTM	0.747	0.619	0.677
BiLSTM	0.784	0.670	0.723
BiLSTM-CRF	0.840	0.808	0.824
CNN-BiLSTM	0.855	0.797	0.825
CNN-BiLSTM-CRF	0.875	0.824	0.849
GC-BiLSTM-CRF	0.894	0.880	0.887

aptability of the model to on-board fault information. Meanwhile, the gated convolution is used to retain the non-linear operation ability and filter useless information through the gating unit. It shows that the gating convolution layer can capture the character representation features well. Therefore, for entities with lower and upper cases, special punctuation marks, and unclear boundaries, the relevant features can be fully obtained to improve the recognition effect.

3.4 Additional features effect verification

In order to verify the effect of additional features on improving the recognition task of on-board fault information, the character embedding without additional features is passed into the proposed model for the experiment. The baseline model is also used in additional feature verification experiments. In this group of experiments, the character embedding and additional features are concatenated as the input of each baseline model. The recognition recall and F1 score of each model with and without additional features are shown in Figs. 3 and 4.

It can be seen from the comparison results that the model with the combination of semantic features and additional features as input is superior to the model with single character embedding as input in the evaluation metrics of recall and F1 score. For example, compared with the model without additional features, the multi-neural network collaboration model with additional features increases 0.027 and 0.016 in recall and F1 score, respectively. CNN-BiLSTM-CRF model ranks second in recognition effect, improved from 0.824 to 0.856 in the recall, and improved from 0.849 to 0.861 in the F1 score.

The results show that the entity boundary can be distinguished effectively by tagging some specific keywords and features, and the recognition effect of each model for on-board fault information is improved. It also shows the value and feasibility of introducing additional features into the recognition model. Therefore, different types of entity features should be fully extended and mined in named entity recognition.

4 Conclusions

This paper investigates the fault information recognition of high-speed railway on-board equipment and proposes a multi-neural network collaboration recognition model based on GC-BiLSTM-CRF. Based on the study of the critical information of on-board faults, the classification rules of named entities in this field are established. The entities are labeled based on the character granularity, improving the quality of data set construction, thus avoiding the error propagation caused by the Chinese word segmentation errors. In order to improve the quality of feature representation of railway domain information, combined with the language characteristics of on-board fault records, a construction method of semantic features and additional features based on character granularity is proposed. The combination of these two features can improve the rationality judgment of fault in-

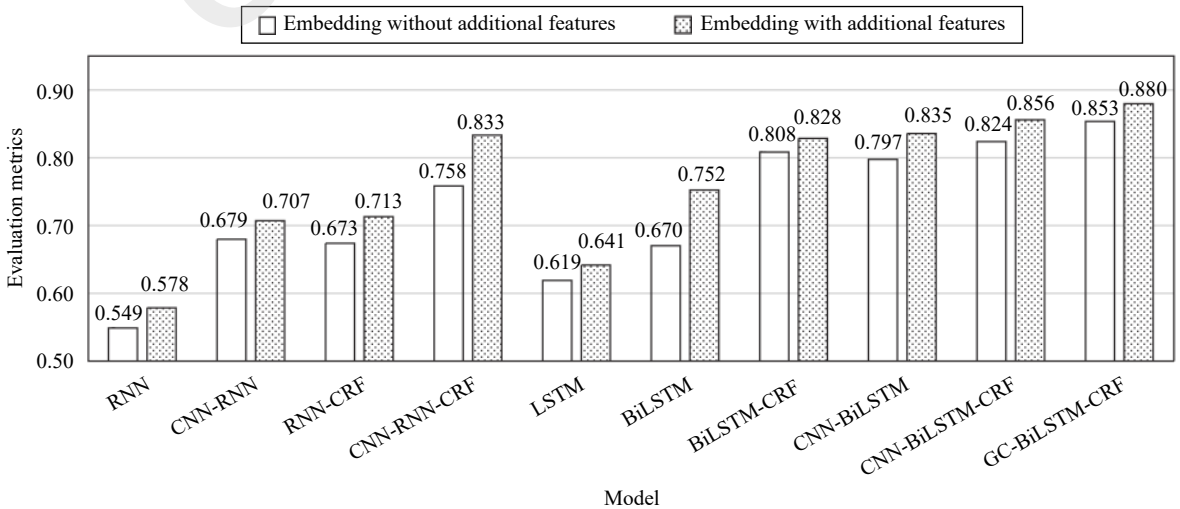

Fig. 3 Recall results of each model

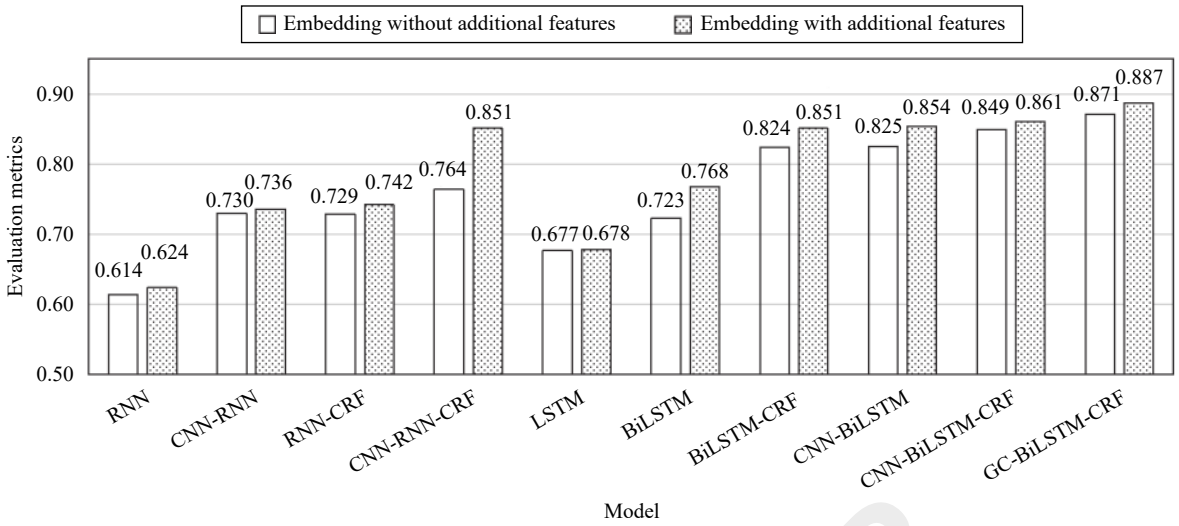

Fig. 4 F1 score results of each model

formation entities. In the construction of additional features, the distinction of entity boundaries can be improved by adding additional functional labels to the unique keywords in the railway field. In the model, gated convolution and BiLSTM are used to learn the inner features of sentences jointly. First, local hierarchical features between adjacent characters are extracted from multiple subspaces in parallel using a gated convolutional layer to avoid the destruction of text sequence information by down-sampling and alleviating the vanishing gradient problem. Then, the forward and backward hidden states of BiLSTM are used to extract past and future information to capture the long-distance dependence of context features. In the output of entity tags, the CRF layer uses the dependency and constraint relationship between adjacent tags to jointly decode the optimized tag sequence of the whole sentence to complete the task of fault information recognition of vehicle equipment.

The GCNN-BiLSTM-CRF model parameters are selected through experiments and compared with several representatives named entity recognition task models, including RNN, LSTM, CNN and their variants. The experimental results show that the proposed model achieves excellent precision, recall, and F1 score. It also proves that the combination of semantic features and additional features can improve the quality of feature representation and further improve the model's recognition effect of the model for on-board fault information.

Acknowledgments

This work was supported by National Natural Science Foundation of China (No. 61763025), Gansu Science and Technology Program Project (No. 18JR3RA104), Industrial Support Program for Colleges and Universities in Gansu Province (No. 2020C-19), and Lanzhou Science and Technology Project (No. 2019-4-49).

References

[1] J. Tekli. An overview on XML semantic disambiguation from unstructured text to semi-structured data: Background, applications, and ongoing challenges. *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 6, pp. 1383–1407, 2016. DOI: 10.1109/TKDE.2016.2525768.

[2] X. F. Mu, W. Wang, A. P. Xu. Incorporating token-level dictionary feature into neural model for named entity recognition. *Neurocomputing*, vol. 375, pp. 43–50, 2020. DOI: 10.1016/j.neucom.2019.09.005.

[3] F. Li, M. S. Zhang, B. Tian, B. Chen, G. H. Fu, D. H. Ji. Recognizing irregular entities in biomedical text via deep neural networks. *Pattern Recognition Letters*, vol. 105, pp. 105–113, 2018. DOI: 10.1016/j.patrec.2017.06.009.

[4] X. Z. Yin, H. Zhao, J. B. Zhao, W. W. Yao, Z. L. Huang. Multi-neural network collaboration for Chinese military named entity recognition. *Journal of Tsinghua University (Science and Technology)*, vol. 60, no. 8, pp. 648–655, 2020. DOI: 10.16511/j.cnki.qhdxxb.2020.25.004. (in Chinese)

[5] L. Luo, Z. H. Yang, P. Yang, Y. Zhang, L. Wang, H. F. Lin, J. Wang. An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition. *Bioinformatics*, vol. 34, no. 8, pp. 1381–1388, 2018. DOI: 10.1093/bioinformatics/btx761.

[6] D. M. Li, Y. Zhang, D. Y. Li, D. Q. Lin. Review of entity relation extraction methods. *Journal of Computer Research and Development*, vol. 57, no. 7, pp. 1424–1448, 2020. DOI: 10.7544/issn1000-1239.2020.20190358. (in Chinese)

[7] R. Bharathi, R. Selvarani. Hidden Markov model approach for software reliability estimation with logic error. *International Journal of Automation and Computing*, vol. 17, no. 2, pp. 305–320, 2020. DOI: 10.1007/s11633-019-1214-7.

[8] Z. Chen, H. Ji. Language specific issue and feature exploration in Chinese event extraction. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, ACM, Boulder, USA, pp. 209–

212, 2009. DOI: 10.3115/1620853.1620910.

[9] G. Luo, X. J. Huang, C. Y. Lin, Z. Q. Nie. Joint entity recognition and disambiguation. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, pp. 879–888, 2015. DOI: 10.18653/v1/D15-1104.

[10] Z. H. Zheng, W. B. Wu, X. Chen, R. X. Hu, X. Liu, P. Wang. A Traffic sensing and analyzing system using social media data. *Acta Automatica Sinica*, vol. 44, no. 4, pp. 656–666, 2018. DOI: 10.16383/j.aas.2017.c160537. (in Chinese)

[11] R. F. He, S. Y. Duan. Joint Chinese event extraction based multi-task learning. *Journal of Software*, vol. 30, no. 4, pp. 1015–1030, 2019. DOI: 10.13328/j.cnki.jos.005380. (in Chinese)

[12] X. Z. Ma, F. Xia. Unsupervised dependency parsing with transferring distribution via parallel guidance and entropy regularization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, USA, pp. 1337–1348, 2014.

[13] Y. M. Han, N. Ding, Z. Q. Geng, Z. Wang, C. Chu. An optimized long short-term memory network based fault diagnosis model for chemical processes. *Journal of Process Control*, vol. 92, pp. 161–168, 2020. DOI: 10.1016/j.jprocont.2020.06.005.

[14] X. Hu, Y. M. Han, B. Yu, Z. Q. Geng, J. Z. Fan. Novel leakage detection and water loss management of urban water supply network using multiscale neural networks. *Journal of Cleaner Production*, vol. 278, Article number 123611, 2021. DOI: 10.1016/j.jclepro.2020.123611.

[15] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, P. Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.

[16] Z. H. Huang, W. Xu, K. Yu. Bidirectional LSTM-CRF models for sequence tagging, [Online], Available: https://arxiv.org/abs/1508.01991, Aug 9, 2015.

[17] J. P. C. Chiu, E. Nichols. Named entity recognition with bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 357–370, 2016. DOI: 10.1162/tacl_a_00104.

[18] Z. G. Liu, X. R. Chen. Research on relation extraction of named entity on social media in smart cities. *Soft Computing*, vol. 24, no. 15, pp. 11135–11147, 2020. DOI: 10.1007/s00500-020-04742-w.

[19] X. Y. Li, Y. X. Meng, X. F. Sun, Q. H. Han, A. Yuan, J. W. Li. Is word segmentation necessary for deep learning of Chinese representations? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, ACL, Florence, Italy, pp. 3242–3252, 2019. DOI: 10.18653/v1/P19-1314.

[20] Q. Zhao, D. Wang, S. S. Xu, X. T. Zhang, X. X. Wang. A weakly supervised Chinese medical named entity recognition method based on RNN. *Journal of Harbin Engineering University*, 2020. (in Chinese)

[21] L. Ratinov, D. Roth. Design challenges and misconceptions in named entity recognition. In *Proceedings of the 13th Conference on Computational Natural Language Learning*, ACM, Boulder, USA, pp. 147–155, 2009.

[22] J. Wang. M. Wang. P. P. Li, L. Q. Liu, Z. Q. Zhao, X. G. Hu, X. D. Wu. Online feature selection with group structure analysis. *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 11, pp. 3029–3041, 2015. DOI: 10.1109/TKDE.2015.2441716.

[23] H. Reddy, N. Raj, M. Gala, A. Basava. Text-mining-based fake news detection using ensemble methods. *International Journal of Automation and Computing*, vol. 17, no. 2, pp. 210–221, 2020. DOI: 10.1007/s11633-019-1216-5.

[24] Y. Bengio, H. Schwenk, J. S. Senécal, F. Morin, J. L. Gauvain. Neural probabilistic language models. *Innovations in Machine Learning*, D. E. Holmes, L. C. Jain, Ed., Berlin, Heidelberg: Springer, pp. 137–186, 2006. DOI: 10.1007/3-540-33486-6_6.

[25] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, Lake Tahoe USA, pp. 3111–3119, 2013.

[26] L. C. Li, Z. Y. Wu, M. X. Xu, H. Meng, L. H. Cai. Combining CNN and BLSTM to extract textual and acoustic features for recognizing stances in mandarin ideological debate competition. In *Proceedings of the Interspeech 2016*, San Francisco, USA, pp. 1392–1396, 2016. DOI: 10.21437/Interspeech.2016-324.

[27] X. L. Tang, W. X. Lin, Y. M. Du, T. Wang. Short text feature extraction and classification based on serial-parallel convolutional gated recurrent neural network. *Advanced Engineering Sciences*, vol. 51, no. 4, pp. 125–132, 2019. DOI: 10.15961/j.jsuese.201801160. (in Chinese)

[28] Y. L. Jin, J. F. Xie, W. S. Guo, C. Luo, D. J. Wu, R. Wang. LSTM-CRF neural network with gated self attention for Chinese NER. *IEEE Access*, vol. 7, pp. 136694–136703, 2019. DOI: 10.1109/ACCESS.2019.2942433.

[29] Y. N. Dauphin, A. Fan, M. Auli, D. Grangier. Language modeling with gated convolutional networks, [Online], Available: https://arxiv.org/abs/1612.08083v3, 2017.

[30] X. Glorot, Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. *Journal of Machine Learning Research*, vol. 9, pp. 249–256, 2010.

[31] R. Pascanu, T. Mikolov, Y. Bengio. On the difficulty of training Recurrent Neural Networks. In *Proceedings of the 30th International Conference on Machine Learning*, Atlanta, USA, pp. 1310–1318, 2013.

[32] S. Hochreiter, J. Schmidhuber. Long short-term memory. *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. DOI: 10.1162/neco.1997.9.8.1735.

Lu-Jie Zhou received the B. Sc. degree in traffic information engineering & control from Lanzhou Jiaotong University, China in 2015. She is currently a Ph. D. degree candidate in traffic information engineering & control from Lanzhou Jiaotong University, China.

Her research interests include intelligent fault diagnosis and natural language processing.

E-mail: 792321186@qq.com (Corresponding author)

ORCID: 0000-0003-4808-6942

Jian-Wu Dang received the Ph. D. degree in electrification & automation of railway traction from Southwest Jiaotong University, China in 1996. He is a professor, doctoral supervisor, vice president of Lanzhou Jiaotong University, China. He is a national candidate for the New Century Ten Million Talent Project and one of the first batch of Special Science and Technology Experts in Gansu Province. He is an expert with outstanding contributions from the Ministry of Railways and won the 6th Zhan Tianyou Railway Science and Technology Award. He has published 5 monographs and published more than 170 academic papers.

His research interests include intelligent information processing, intelligent transportation, and image processing.

E-mail: dangjw@mail.lzjtu.cn

Zhen-Hai Zhang received the Ph. D. degree in traffic information engineering & control from Lanzhou Jiaotong University, China in 2014. He is an associate professor, master supervisor of Lanzhou Jiaotong University, China. He has published 14 relevant academic papers and participated in the compilation of 2 teaching materials.

His research interest is intelligent transportation.

E-mail: zhangzhenhai@lzjtu.cn