A deep learning method for named entity recognition in bidding document

Yunfei Ji¹, Chao Tong², Jun Liang³, Xi Yang³, Zheng Zhao³ and Xu Wang¹,*

¹College of Computer Science, Sichuan University, 610065 Chengdu Sichuan, China
²China Mobile (Suzhou) Software Technology Co., Ltd, 215000 Suzhou Jiangsu, China
³Corresponding author’s e-mail: shawnwongmilab@gmail.com

Abstract. Named entity recognition is a fundamental task of Natural Language Processing and belongs to the category of sequence labeling problems. In the text, the named entity is the main carrier of information, which is used to express the main content of the text. Accurately identifying these contents is essential for implementing various natural language processing techniques such as information extraction, information retrieval, machine translation, and question and answer system. Named entities in business documents contain a lot of important information that can bring significant business value to the business. In this paper, we propose the method of combining Bi-directional long-short-term memory network and conditional random field, combining n-gram features and character features, and introducing attention mechanism to identify the tenderee, bidding agent and project number three entities in the bidding documents. Compared with the LSTM, BiLSTM can obtain the context information better and extract more features. The CRF uses the features obtained by BiLSTM to decode and obtain the final labeling result. In the comparative experiment of the collected data sets of 20,000 bidding documents, the BiLSTM-CRF model proposed in this paper can produce better labeling effect than other models and meet our expectations.

1. Introduction

Named entity recognition is one of the basic tasks of natural language processing[1], referring to the identification of entities with specific meaning in the text, including person names, place names, institution names, proper nouns, etc. It is an important part of information extraction, machine translation, text classification, question and answer system, text summary, and intelligent search. In the text, the named entity is the main carrier of information, used to express the main content in the text, so named entity recognition is very important for extracting key information of the text. In a narrow sense, it refers to a named entity recognition means recognizes the name of person, place and organization name three types of entities, but in a specific field, different entity types within the domain are defined accordingly.

Named entity recognition is very helpful for the extraction of information in business documents. For example, the extraction of winning information in a tender is very valuable to the company. It is necessary to identify important information such as the tenderer, bidding agent, and project number in the tender. The tender is the written material provided by the tenderer to the bidder for the bidding work. The information contained in the tender has great commercial value for the enterprise. Collecting a large amount of tender text and screening and identifying important information contained therein is an important business intelligence for enterprises.
Currently, the main technical methods for named entity recognition are rule-based methods, statistical-based methods and deep learning methods. In this paper, we use the method of combining Bidirectional Long-Short-Term Memory Network[4] and Conditional Random Field[5]. In particular, we introduce n-gram vectors into the input vector[6] and added the attention mechanism.

2. Related Works

2.1 Rule-based Methods
Rule-based methods often use linguistic experts to manually construct rule templates, including features such as statistical information, punctuation, keywords, pointers and direction words, position words (such as tail words), and central words, such as patterns and strings. Matching is the main means, and most of these systems rely on the establishment of knowledge bases and dictionaries. Rule-based and dictionary-based methods are the earliest methods used in named entity recognition. In general, rule-based methods perform better than statistical-based methods when extracted rules can more accurately reflect linguistic phenomena. However, these rules often depend on the specific language, domain and text style. The compilation process is time-consuming and difficult to cover all linguistic phenomena. It is especially prone to errors, system portability is not good, and linguistic experts need to rewrite rules for different systems. Another shortcoming of the rule-based approach is that it is too costly, and there are problems such as long system construction period, poor portability, and the need to establish different domain knowledge bases as an aid to improve system identification capabilities.

2.2 Statistical-based Methods
The methods based on statistical machine learning mainly include Hidden Markov Model[2][8], Maximum Entropy[7], Support Vector Machine[9], Conditional Random Field[10] and so on. The statistical-based method has high requirements for feature selection. It is necessary to select various features that affect the task from the text and add these features to the feature vector. According to the main difficulties and characteristics of the specifically named entity identification, consider selecting a feature set that can effectively reflect the characteristics of the entity. The main method is to dig out the characteristics from the training corpus by statistically analyzing and analyzing the linguistic information contained in the training corpus. Relevant features can be divided into specific word features, context features, dictionary and part-of-speech features, stop word features, core word features, and semantic features. The statistical-based approach relies heavily on the corpus, and the large-scale general corpus that can be used to construct and evaluate named entity recognition systems is relatively small. The text in the model training process requires a lot of human participation in the markup, and the cost is too high.

2.3 Deep Learning Methods
In recent years, Deep Learning has performed well in many fields of natural language processing, such as part-of-speech tagging, text segmentation, intelligent question answering, and sentiment analysis. More and more researchers also use deep learning to deal with named entity recognition[3]. In this paper, we use the method of combining Bidirectional LSTM and CRF, which is BiLSTM-CRF[11], to identify three named entities: tenderee, bidding agent and project number. Feature extraction is performed on the input sequence using a BiLSTM. The effect of using BiLSTM is better than that of unidirectional LSTM, because BiLSTM traverses both forward and reverse sequences, and more features can be extracted than unidirectional LSTM. After passing through the BiLSTM layer, we introduce the attention mechanism, which assigns a higher weight to the more important part of the named entity identification, and then uses a CRF layer to take the features extracted from the BiLSTM layer as input and to calculate the label of each element in the sequence.

In particular, we introduce n-gram vectors into the input vector and added the attention mechanism. Using both n-gram features and character features can obtain more information. Furthermore, the
attention mechanism can assign different weights to features of different importance levels, which can significantly improve the recognition effect of named entities.

3. The Proposed Method

3.1 Neural Network Architecture

Based on the existing BiLSTM-CRF network[22], this paper aims to improve the network model. The n-gram and character features are used as the input of the network model. The bidirectional expression of the text context information is obtained through the BiLSTM network, and the attention is added to distinguish importance.

Figure 1 is a structural diagram of the BiLSTM-CRF network proposed in this paper, which is mainly divided into five layers: a vector representation layer, a BiLSTM layer, a feature fusion layer, an attention layer, and a CRF layer. The input of the network is a sequence of texts, and the output is a sequence of tokens corresponding to the text. After the original text sequence is vectorized, input into the BiLSTM network and BiLSTM automatically obtains the text information before and after the current word through the training process.

After the BiLSTM network obtains the context information of the current text, an attention layer is used to assign different weights to different features. Then merge the n-gram features and character features as input to the CRF layer. The CRF layer outputs the optimal label sequence of the current sequence according to the feature information acquired by the BiLSTM.

![Figure 1. A structural diagram of the BiLSTM-CRF network proposed in this paper. The input shown in Figure 1. are the Pinyin form of Chinese characters.](image)

3.2 Detail of the Model

3.2.1 Embedding Layer. Compared with the English text composition, as the basic composition granularity of Chinese, a single Chinese character has rich semantic information and can also be used for modeling. On the basis of adding the character features, this paper further fuses the n-gram features and enhances the semantic feature information of the named entity recognition.

The most intuitive and the most commonly used word representation in NLP is One-hot representation[16], which represents each word as a very long vector. The dimension of this vector is
the vocabulary size, where most of the elements are 0, and only one dimension has a value of 1, which represents the current word.

However, an important problem with this representation is the “word gap” phenomenon, any two words are isolated. It is not obvious from these two vectors whether the two words have a relationship, which destroys the semantic connection between words and words. The distribution representation effectively solves these problems. It can express the similarity between words and words, and contains more information, solving the problem of excessive storage space.

In this paper, we use the pre-trained n-gram vector and character vector presented by Li S, et al.

### 3.2.2 BiLSTM

Recurrent Neural Networks have achieved great success and wide application in many NLP problems. Cho K, et al. used RNN in statistical machine translation. In theory, RNN can process sequence data of any length.

An RNN network can be represented as Figure 2, including the input layer \( x \), the hidden layer \( h \), and the output layer \( y \), where \( x \) represents the input characteristics and \( y \) represents the type of the output tag, using the BIOES notation. The specific calculation method is as follows:

\[
\begin{align*}
  h(t) &= F\left( Ux(t) + Wh(t-1) + b_h \right) \\
  y(t) &= G\left( Vh(t) + b_y \right)
\end{align*}
\]

Where, \( F \) is the hidden layer activation function and \( G \) is the Softmax function.

![Figure 2. Simple structure diagram of RNN. The input shown in Figure 2. are the Pinyin form of Chinese characters.](image)

When some specific tasks have only a special relationship with similar moments, the RNN model performs very well, but when the two distant moments are also related, the RNN cannot be expressed, and there may be a problem with the gradient disappearing. LSTM is generally employed to solve the above problem.

LSTM is a special type of RNN that can learn long-term dependencies and is widely used when processing sequence data.[19]

The core of LSTM is the cell state, which is the right line in Figure 3. The state of the cell is like a conveyor belt, its state is transmitted along the entire chain, and only a few places have some linear interaction. If the information is passed in this way, it will actually remain the same. The LSTM controls the state of the cell through a structure called a "gate" and truncates or adds information to it. LSTM has three such gates: the forgetting gate, the input gate and the output gate, which control the state of the cell. Its operation can be expressed as follows:

\[
\begin{align*}
  i_t &= \sigma(W_{ix} x_t + W_{ih} h_{t-1} + W_{ic} c_{t-1} + b_i) \\
  f_t &= \sigma(W_{fx} x_t + W_{fh} h_{t-1} + W_{fc} c_{t-1} + b_f) \\
  c_t &= f_t c_{t-1} + i_t \tanh(W_{xc} x_t + W_{hc} h_{t-1} + b_c)
\end{align*}
\]
\[ o_t = \sigma(W_{xo} x_t + W_{ho} h_{t-1} + W_{co} c_{t-1} + b_o) \]  \hspace{1cm} (6) \\
\[ h_t = o_t \tanh(c_t) \]  \hspace{1cm} (7) 

Where \( \sigma \) is a Sigmoid function, and the flags \( i, f, o \) and \( c \) are activation vectors representing Input Gate, Forget Gate, Output Gate, and Memory Cell, respectively. In a text sequence, for the \( t \) th character, the inputs of LSTM are \( x_t, c_{t-1} \) and \( h_{t-1} \), and their corresponding outputs \( c_t \) and \( h_t \) can be calculated according to the above formula.

\[ x_1, x_2, \ldots, x_n \]

In this paper, we use BiLSTM to solve the problem of named entity recognition in the tender. LSTM has a huge advantage in dealing with text sequences, since it can obtain long-distance information dependencies. BiLSTM can obtain more comprehensive context information. The basic idea of Bi-LSTM is to propose that each training sequence is two LSTMs forward and backward, and both are connected to an output layer. For the input sequence \( X = \{x_1, x_2, \ldots, x_n\} \), the forward LSTM receives the input sequence in the order of \( x_1 \) to \( x_n \), calculates the forward hidden state \( \tilde{h} \), and the reverse LSTM receives the input sequence in the order of \( x_n \) to \( x_1 \), calculates the reverse hidden state \( \hat{h} \). By concatenating the forward hidden state with the backward hidden state, we can get the complete past and future context information \( \tilde{h} \) for each point in the output layer input sequence, as shown in Figure 4.

\[ F_{in} = [F_{in}; F_{out}] \]  \hspace{1cm} (8) 

3.2.3 Feature Fusion Layer. The feature fusion layer mainly integrates the character features and n-gram features, and uses it as the overall input feature of the attention layer. The fusion method is as shown in equation (8):

\[ F_{in} = [F_{in}; F_{out}] \]  \hspace{1cm} (8)
Among them, \( F_c \) is the character features output of the attention layer, \( F_n \) is the n-gram features and \( F_{cn} \) is the feature of fusion.

### 3.2.4 Attention Layer

The visual attention mechanism is a brain signal processing mechanism unique to human vision. By quickly scanning the global image, human vision obtains the target area that needs to be focused on and then invests more attention resources in this area to obtain more detailed information about the target.

There is a problem with the traditionally structured LSTM/RNN model: it is encoded into a fixed-length vector representation regardless of the length of the input, which makes the model poorly learning for long input sequences. The attention mechanism overcomes the above problem. The principle is to selectively focus on the corresponding information in the input when the model is output.

The attention mechanism[13] was first applied to the field of the visual image. In recent years, the neural network based on the attention mechanism has gradually become a hot topic in deep learning research. The attention mechanism method is widely used in various sequence prediction tasks, including text translation, speech recognition, etc. For the first time, Bahdanau[14] applied the attention mechanism to the NLP field. They applied the attention mechanism to the machine translation task, enabling simultaneous translation and alignment. Since then, various deep learning networks that incorporate attention mechanisms have gradually expanded into various NLP tasks. Yang et al. [15] proposed a multi-layered attention mechanism network for text categorization. By using attention mechanisms for different features in the text, the importance of various words and sentences in the text is determined, and the effect of text categorization is improved.

In the bidding text, there are some parts that are more important for the identification of named entities. If these parts can be marked, the recognition effect of the named entities will inevitably increase. The attention mechanism is a good way to distinguish the importance of the current input target unit and other units. This paper uses attention layer to focus on the character features and the n-gram features so that more weights are assigned to the valuable parts.

As shown in Equation 8, \( \hat{h}_t \) is the BiLSTM hidden layer output, \( W_i \) is the weight, \( b_i \) is the bias term, and \( \tanh \) is the activation function.

\[
Y = \tanh(W_i \hat{h}_t + b_i) \tag{9}
\]

The output \( Y \) is then subjected to the softmax function to obtain the probability of each category, which is the attention vector matrix, indicating the importance of the hidden state at that moment:

\[
\alpha_i = \text{soft max}(Y_i) = \exp^Y_i / \sum_{j=1} \exp^Y_j \tag{10}
\]

Finally, a weighted summation is obtained to obtain the attention output vector:

\[
V = \sum_{i=1}^t \alpha_i \hat{h}_i \tag{11}
\]

The attention mechanism, according to \( \alpha \), can treat the state of the hidden layer at each moment differently.

### 3.2.5 CRF

CRF has been used for sequence labeling problems long ago, but feature extraction in the process requires manual design and easy loss of emotional information. The context information automatically obtained by BiLSTM can be directly used as the input of the CRF after the different weights are assigned by the attention layer. Compared with the labeling of the single character by the LSTM output, the CRF can obtain the dependency between the adjacent tags. Therefore, it is possible to largely avoid the illegal marking order (for example, \texttt{B-ORG} cannot appear in front of the \texttt{I-ORG}, etc.), and the recognition effect is greatly improved.
4. Experiments and Results

4.1 Dataset
We collected 20,000 bidding documents, which have been marked with tenderer, bidding agent, and project number entities for easy testing. It was randomly divided into a training set and a test set, containing 15,000 data and 5,000 data respectively. For the annotation method of named entities, we use the most common BIOES annotation rules to mark, where B means that the character is at the beginning of an entity, I means inside, O means outside, E means the character is at the end of an entity, and S means that the character itself can form an entity.

4.2 Model Setting
We implement the entire network structure through the NCRF++ framework proposed in [17]. NCRF++ is a PyTorch-based framework with the flexible choice of input functions and output structures. NCRF++ is divided into three layers, Character sequence representation, Word sequence representation, and Inference layer. Each layer can be customized by corresponding parameter settings. NCRF++ supports different structure combinations of three levels, no coding, and has good performance.

The n-gram vector and character vector are both 300-dimensional, batch size is set to 32, iteration is set to 50, and dropout is 0.5. We use mini-batch stochastic gradient descent (SGD) with a decayed learning rate to update parameters. In addition to the above settings, we performed five comparative experiments on the above dataset by changing the parameter settings using the following models, the sixth experiment below used the model proposed in this paper.

1) CNN. Named entity recognition is performed in the above dataset by a convolutional neural network method.
2) LSTM. Named entity recognition is performed on the above dataset using a unidirectional LSTM network, using only character vectors as input features.
3) BiLSTM. Named entity recognition is performed on the above dataset using a BiLSTM network, using only character vectors as input features.
4) BiLSTM+CRF. Named entity recognition is performed on the above dataset using the classic BiLSTM+CRF network, using only character vectors as input features.
5) CLSTM+WLSTM+CRF. Named entity recognition is performed in the above dataset using the BiLSTM+CRF network, and both the n-gram vector and character vector are used as input for the named entity recognition, and no attention is added.
6) CLSTM+WLSTM+CRF(with attention). Use both the n-gram vector and character vector as input to perform named entity recognition, add attention layer.

4.3 Results
Experiment with the existing dataset using the above six methods. Precision, recall and F1-score were used as evaluation criteria. Table 1-3 gives the experimental results of three types of named entities.

| Models               | Tenderee | Bidding Agent | Project Number |
|----------------------|----------|---------------|----------------|
| CNN                  | 85.69    | 85.12         | 86.21          |
| LSTM                 | 87.65    | 85.36         | 86.46          |
| BiLSTM               | 85.67    | 86.55         | 87.02          |
| BiLSTM+CRF           | 86.38    | 86.21         | 86.35          |
| CLSTM+WLSTM+CRF      | 89.24    | 89.34         | 86.37          |
Table 2. Comparison of recall from experiments using the above six models

| Models                        | Tenderee | Bidding Agent | Project Number |
|-------------------------------|----------|---------------|----------------|
| CNN                           | 86.69    | 85.04         | 83.70          |
| LSTM                          | 87.23    | 87.34         | 84.11          |
| BiLSTM                        | 88.52    | 89.51         | 86.34          |
| BiLSTM+CRF                    | 88.67    | 88.41         | 85.35          |
| CLSTM+WLSTM+CRF (with attention) | 89.84    | 89.60         | 86.53          |
| CLSTM+WLSTM+CRF (with attention) | 91.85    | 91.34         | 86.73          |

Table 3. Comparison of F1-score from experiments using the above six models

| Models                        | Tenderee | Bidding Agent | Project Number |
|-------------------------------|----------|---------------|----------------|
| CNN                           | 86.19    | 85.08         | 84.94          |
| LSTM                          | 87.44    | 86.34         | 85.27          |
| BiLSTM                        | 87.07    | 88.01         | 86.68          |
| BiLSTM+CRF                    | 87.51    | 87.30         | 85.85          |
| CLSTM+WLSTM+CRF               | 89.54    | 89.47         | 86.45          |
| CLSTM+WLSTM+CRF (with attention) | 91.28    | 90.24         | 86.52          |

5. Conclusion
In this paper, we use a deep learning method to identify the tenderer, bidding agents, and project number three named entities in the tender. We use the BiLSTM-CRF to obtain semantic and morphological information. The key is that we introduce the n-gram vector and fuse it with the character vector. After vectorizing the input text, we obtain the bidirectional expression containing the context information through the BiLSTM network and then innovatively use the attention mechanism. The parts with different importance levels are given different weights. By inputting the obtained context features into the CRF, the optimal mark sequence for the current input sequence is obtained. Compared with the direct use of the Softmax classification, the CRF can obtain the dependence between adjacent tags to avoid the occurrence of illegal marking order, effectively improve the effect of named entity recognition, we also verified the improvement of experimental results by the character vector after the fusion n-gram feature through contrast experiments. The experimental results show that in the real bidding text dataset, the BiLSTM-CRF network proposed in this paper, with the fusion of n-gram vector and character vector as input and added the attention mechanism, has a good effect and basically meets the needs of enterprises.

Acknowledgments
This research was supported by the Chinese Education Department-China Mobile Research Foundation (Grant Number: MCM20170309).
References

[1] Zitouni, I. (2014) Natural language processing of semitic languages. Springer, Berlin.
[2] Zhou, G.D., Su, J. (2002) Named entity recognition using an HMM-based chunk tagger. In: proceedings of the 40th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics. Philadelphia. pp. 473-480.
[3] Huang, Z., Xu, W., Yu,K. (2015) Bidirectional LSTM-CRF models for sequence tagging. Computer Science.
[4] Hochreiter, S., Schmidhuber, J. (1997) Long short-term memory. Neural computation, 9(8): 1735-1780.
[5] Sutton, C., McCallum, A. (2012) An introduction to conditional random fields. Foundations and Trends® in Machine Learning, 4(4): 267-373.
[6] Mikolov, T., Chen, K., Corrado, G., et al. (2013) Efficient estimation of word representations in vector space.
[7] Berger, A. L., Pietra, V. J. D., Pietra, S. A. D. (1996) A maximum entropy approach to natural language processing. Computational linguistics,22(1): 39-71.
[8] Bikel, D. M., Miller, S., Schwartz, R., et al. (1997) Nymble: a high-performance learning name-finder. In: The fifth conference on Applied natural language processing. Washington, DC. pp. 194-201.
[9] Hearst, M. A., Dumais, S. T., Osuna, E., et al. (1998) Support vector machines. IEEE Intelligent Systems and their applications, 13(4): 18-28.
[10] Sutton, C., McCallum, A. (2012) An introduction to conditional random fields. Foundations and Trends® in Machine Learning, 4(4): 267-373.
[11] Lin, B. Y., Xu, F., Luo, Z., et al. (2017) Multi-channel bi-lstm-crf model for emerging named entity recognition in social media. In: The 3rd Workshop on Noisy User-generated Text. Copenhagen. pp. 160-165.
[12] Goodfellow, I., Bengio, Y., Courville, A., et al. (2016) Deep learning. MIT press, Cambridge.
[13] Vaswani, A., Shazeer, N., Parmar, N., et al. (2017) Attention is all you need. In: Advances in Neural Information Processing Systems. pp. 5998-6008.
[14] Bahdanau, D., Cho, K., Bengio, Y. (2014) Neural machine translation by jointly learning to align and translate. Computer Science.
[15] Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E. (2017) Hierarchical Attention Networks for Document Classification. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp.1480-1489.
[16] Turian, J., Ratinov, L., & Bengio, Y. (2010) Word representations: a simple and general method for semi-supervised learning. In: The 48th annual meeting of the association for computational linguistics. Uppsala. pp. 384-394.
[17] Yang, J., Liang, S., Zhang, Y. (2018) Design challenges and misconceptions in neural sequence labeling.
[18] Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., Khudanpur, S. (2015) Recurrent neural network based language model. INTERSPEECH 2010, Conference of the International Speech Communication Association. Chiba. pp.1045-1048.
[19] Dong, C., Zhang, J., Zong, C., Hattori, M., Di, H. (2016) Character-Based LSTM-CRF with Radical-Level Features for Chinese Named Entity Recognition. Springer International Publishing, Berlin.
[20] Li, S., Zhao, Z., Hu, R., Li, W., Liu, T., Du, X. (2018) Analogical reasoning on chinese morphological and semantic relations.
[21] Cho, K., Merrienboer, B. V., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., et al. (2014) Learning phrase representations using rnn encoder-decoder for statistical machine translation. Computer Science.
[22] Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C. (2016) Neural architectures for named entity recognition. 260-270.