International Journal on Semantic Web and Information Systems
Volume 18 • Issue 1

Chinese Named Entity Recognition Method Combining ALBERT and a Local Adversarial Training and Adding Attention Mechanism

Zhang Runmei, Anhui Jianzhu University, China
Li Lulu, Anhui Jianzhu University, China*
Yin Lei, AnHui JianZhu University, China
Liu Jingjing, AnHui JianZhu University, China
Xu Weiyi, AnHui JianZhu University, China
Cao Weiwei, Key Laboratory of Flight Techniques and Flight Safety, China
Chen Zhong, Anhui Jianzhu University, China

ABSTRACT

For Chinese NER tasks, there is very little annotation data available. To increase the dataset, improve the accuracy of Chinese NER task, and improve the model's stability, the authors propose a method to add local adversarial training to the transfer learning model and integrate the attention mechanism. The model uses ALBERT for migration pre-training and adds perturbation factors to the output matrix of the embedding layer to constitute local adversarial training. BILSTM is used to encode the shared and private features of the task, and the attention mechanism is introduced to capture the characters that focus more on the entities. Finally, the best entity annotation is obtained by CRF. Experiments are conducted on People's Daily 2004 and Tsinghua University open-source text classification datasets. The experimental results show that compared with the SOTA model, the F1 values of the proposed method in this paper are improved by 7.32 and 7.98 in the two different datasets, respectively, proving that the accuracy of the method in this paper is improved in the Chinese domain.

KEYWORDS

Albert, BiLSTM-CRF, Chinese Named Entity Recognition, Deep Learning, Transfer Learning

INTRODUCTION

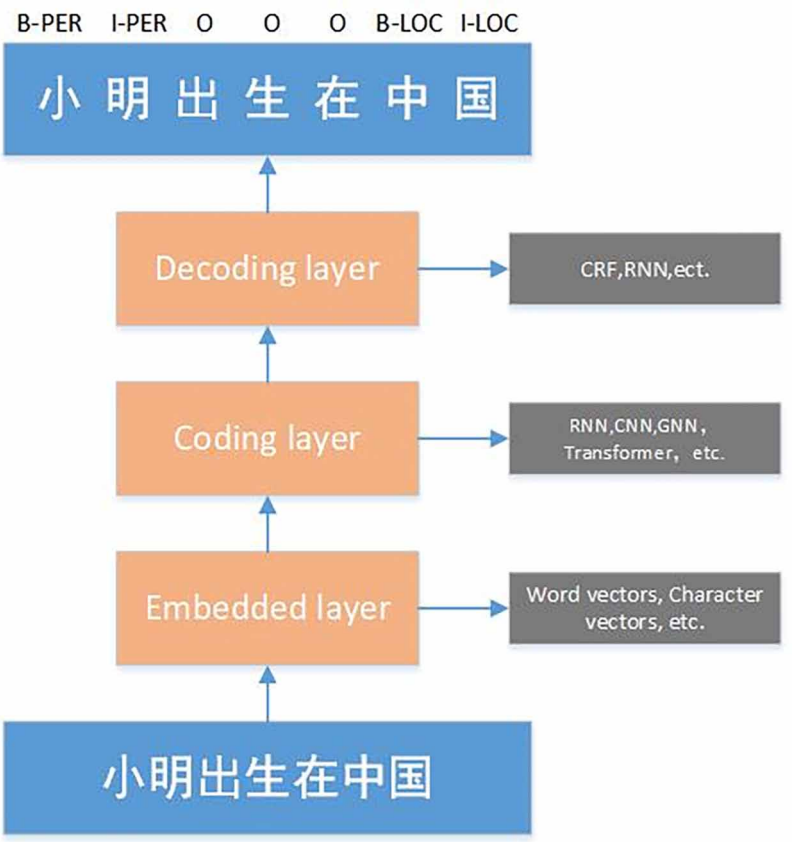
Natural language processing (NLP) has become hot research in the field of artificial intelligence and deep learning. The current challenge is how to combine advanced natural language processing and machine learning models so that machines can understand the learned knowledge, express it as a kind of knowledge, and establish relevant connections (Liu et al., 2022; Mandle et al., 2022).

Named entity recognition (NER) is a fundamental task studied in NLP. Named entity recognition mainly involves extracting words or phrases from unstructured text that reflect concrete or abstract entities that already exist in the real world, such as names of people, places, and organizations, and

DOI: 10.4018/IJSWIS.313946 *Corresponding Author

organizing them into semistructured or structured information. Then, other techniques are used to analyze and understand the text (Isozaki & Kazawa, 2002; Li et al., 2019). In recent years, with deep learning research, this method can actively learn data feature representation from massive data and reduce the reliance on rules and expert knowledge to a certain extent.

Figure 1. Chinese named entity identification flow chart



The flow of the traditional named entity recognition method is shown in Figure 1. Although deep learning methods have made significant progress in Chinese NER tasks, building NER models usually still requires a large amount of labeled data. Model performance is proportional to the amount of labeled data, which is poor in specific domains where the training corpus is scarce. Transfer learning aims to improve the learning performance of the target task by exploiting a large amount of labeled data in the source domain and pretrained models. It has become a powerful tool for solving resource-poor NER with its advantages of small data and labeling dependencies and relaxed independent homogeneous distribution constraints (Gong et al., 2018; Wang et al., 2017).

From the early word vectors obtained by cooccurrence matrix and SVD(Singular Value Decomposition) decomposition methods, neural network models based on different structures have been developed. At present, word vector models have been widely used in information extraction. Mikolov et al. (2013) proposed the Word2Vec model to generate word vectors. Xu et al. (2019) proposed a word vector representation based on the Word2Vec technique to trigger word classification

tasks. However, the obvious drawback of Word2Vec is that it is "one-word-one-sense," and the semantic meaning expressed by its word vectors is homogeneous. The ALBERT model is a lightweight network model developed by Lan et al. (2020) based on the transformer bidirectional encoder representations from transformers (BERT; Agrawal et al., 2022) pretrained language model. Therefore, this model uses ALBERT to obtain context-dependent word vectors. ALBERT has richer semantic information than the traditional Word2Vec model and can generate more appropriate feature representations for different NLP tasks. Thus, model performance is improved, made smaller, and more refined than the BERT model (Bikel et al.,1998).

Table 1. Comparison and Analysis of Existing Named Entity Recognition Methods

Distributed input representation	Method	Advantage	Shortcoming
Word level	Skip-gram, Word2VecvGlove, fastText, ELMo, BERT, GPT	Good description of the similarity between words	There is an OOVproblem, and Chinese named entity recognition has word segmentation error propagation
Character level	CNN, RNN, BILSTM, BERT, ALBERT, BiGRU	Effectively solve the OOV problem and provide morphological feature information	It is difficult to capture semantic information and boundary information at word level
Mixed multifeature	Improve character level CNN/RNN+dictionary information features, BERT and its variants+dictionary, radical information features, Pooled Contextualized Embeddings	Mixed character features, word features, part of speech features, syntactic features, location features, radical information features, pinyin features, domain dictionaries or knowledge maps, and other external information to strongly represent information	Damage the universality of the system

Note. ALBERT = a lite bert ; BERT = bidirectional encoder representations from transformers; BiGRU = bidirectional gated recurrent unit ; BILSTM = bidirectional long short-term memory; CNN = convolutional neural network; ELMo =embeddings from Language models; GPT =generative pre-training ; OOV = out of vocabulary ; RNN = recurrent neural network.

The advantages and disadvantages of existing named entity recognition methods are shown in Table 1. Existing NER methods mostly focus on improving the model structure and entity labeling, with less attention to the problem of boundary sample confusion in NER data sets. By contrast, deep learning combined with adversarial training has become a new trend in text processing. Miyato et al. (2017) added perturbation at the word vector level for the first time semisupervised text classification based on deep learning. Zhou et al. (2019)[REMOVED REF FIELD] added perturbation at the word embedding level to improve the generalization ability of low-resource NER models. Although these methods can handle larger and more complex features, they also affect the efficiency of downstream tasks because of the long process of generating word vectors and a large amount of training data.

To solve the problem of long training time under migration learning and improve the accuracy and robustness of training, this paper proposes a method of adding local adversarial training to transfer

learning model ALBERT. Adversarial training is used to preserve the original data features while improving the robustness and generalization ability of the model in an adversarial attack manner to improve the model's ability to identify confusion boundary samples. Inspired by the idea of difficult sample mining, only difficult samples that are prone to misclassification in the data are added to the perturbation to reduce redundant adversarial samples. The experiments show that the method in this paper provides ideas for future research on NER using deep learning methods in a targeted manner on small Chinese sample data sets.

In summary, the contributions of this paper are as follows:

- We propose a Chinese named entity recognition framework that integrates transfer learning and local adversarial training.
- We introduce the attention mechanism, which can better obtain the structural features in sentences.
- We have conducted experiments on two types of Chinese data sets, and compared with the baseline model, we can see that the accuracy of our model has been greatly improved.

BACKGROUND

The work in this paper deals with two aspects: local adversarial migration training and adding attention mechanisms to the NER approach.

In NER, the discrete data types need to be converted into machine-recognizable vector types first, and several common character vector representations are used: one-hot encoding, integer encoding, and continuous encoding. However, one-hot encoding takes up too much space and integer encoding cannot express the relationship between semantics, so continuous encoding is chosen for character vectors (Li et al., 2020). Continuous character vector encoding can be obtained by models such as Word2Vec (Mikolov et al., 2013), Glove (Pennington et al., 2014), BERT (Dong et al., 2021), and ALBERT (Wang et al., 2017).

BERT is an unsupervised pretraining model for NLP. BERT set a new record for a very large number of data sets at the time and promoted the rapid development of the NLP field. XLNet, proposed by Yang et al. (2019), solved the "Mask" operation in the BERT model by using the sorting language model for human error. It achieved better performance in individual areas, such as reading comprehension. Roberta, proposed by Liu et al. (2019), improved the length of the training sequence, and the performance is somewhat improved over XLNet. Several improved models based on BERT improve it toward the perspective of the continuous heap of data and the number of parameters. To solve the problem of an excessive number of parameters, and improve the phenomenon that the effect of models with too many parameters decreases instead, ALBERT was proposed. It is a lightweight converter-based bidirectional encoding representation model mainly used for self-supervised learning in the field of language representation.

Adversarial training was developed from adversarial generative networks and was initially applied to improve the robustness of image processing models. In the NLP area, Alzantot et al. (2018) proposed a population-based optimization algorithm to find the nearest replacement words to generate perturbations by repeatedly selecting samples of similar target label classes at random. Li et al. (2019) captured important words that were meaningful for classification and then added small perturbations to these to generate adversarial samples to guide a deep learning classifier for misclassification. Gong et al. (2018) used gradient descent to perturb word vectors into target classes as a way to improve the quality of the adversarial text.

Early research on NER often used Support Vector Machine (SVM) (Isozaki & Kazawa, 2002), hidden Markov (HMM), and CRF (Lafferty et al., 2001) models that rely heavily on feature engineering (Li et al., 2020; Liu., 2021). Deep learning has the advantage of handling larger and more complex features. At the beginning, a convolutional neural network (CNN) was widely used in the image field and natural language processing (Mandle et al., 2022). However, due to its shortcomings, such as

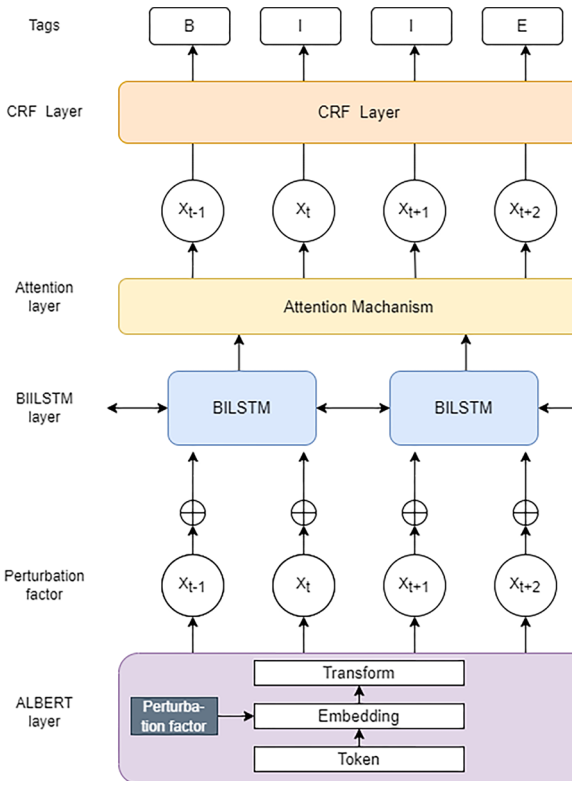
requiring a large number of parameter adjustments, Huang et al. (2015) began to use bidirectional long short-term memory (BiLSTM) for feature extraction and CRF for sequence annotation. Wang et al. (2017) propose a gated convolutional neural network (GCNN) model for Chinese NER.

BiLSTM structure can better capture the bidirectional semantic features of the preceding and following text than can the unidirectional LSTM (Lan et al., 2020). The BiLSTM-CRF model is a benchmark model to solve the text sequence labeling problem. Zhang & Yang (2018) proposed a lattice LSTM model, which obtains all words for encoding by matching characters and splitting words, thereby fully using word and word sequence information. Zhang et al. (2019) proposed a character-level text model based on BiLSTM + CRF for joint training. Zhong et al. (2020) proposed a BiLSTM-CRF model with a self-attentive mechanism, aiming to solve the problem of Chinese word ambiguity and wordless boundaries.

METHOD

The model proposed in this paper mainly obtains word vectors by ALBERT migration training, adds small perturbation factors to the obtained word vectors for local adversarial training, then inputs them to the feature extraction layer, adds an attention mechanism to focus more on local features in the BILSTM layer, and, finally, outputs labeled entities through the conditional random field. It is divided into four parts: embedding layer, feature extraction layer, conditional random field module, and adversarial training. Its overall framework is shown in Figure 2.

Figure 2. The Model Framework Proposed in this Paper: ALBERT+ Local Adversarial Training+BILSTM+Attention+CRF *Note.* **ALBERT = a lite bert ; CRF = Conditional random field; BILSTM = bidirectional long short-term memory; B =Begin; I =inside; O =end ;**



Pretrain the Layer

ALBERT has made the following improvements compared to the mainstream BERT migration model.

- The first improvement is in the factorization of embedding parameterization. The researcher decomposes the large lexical embedding matrix into two smaller matrices, thereby separating the size of the hidden layer from the size of the lexical embedding. Instead of mapping one-hot vectors directly to the hidden space of size H, the researcher first maps them to a low-dimensional word embedding space E and then to the hidden space. The details are shown in Equation (1). By this decomposition, the word embedding parameters can be reduced from $O\left(V \times H\right)$ to $O\left(V \times E + E \times H\right)$, which is a very significant reduction in the number of parameters when H is much larger than E (Haoliang et al., 2019). This separation makes adding hidden layers easier without significantly increasing the number of parameters of the lexical embedding.

$$O\left(V \times H\right) A U \rightarrow O\left(V \times E + E \times H\right) \tag{1}$$

- Second, the technique is cross-layer parameter sharing. This technique avoids the increase in the number of parameters with the increase in network depth. Both techniques significantly reduce the number of parameters in BERT without significantly affecting its performance, thereby improving the parameter efficiency. ALBERT has a similar configuration to BERT-large, but the number of parameters is 1/18th of the latter, and the training speed is 1.7 times faster. ALBERT's transition from one layer to another is much smoother than that of BERT, and weight sharing effectively improves the robustness of the neural network parameters. These parameter reduction techniques can also act as some form of regularization, potentially making the training more stable and facilitating generalization (Yang et al., 2019).
- Third, to further improve the performance of ALBERT, the researchers also introduced a self-supervised loss function for sentence order prediction (SOP), which can significantly improve the performance of downstream multisentence encoding tasks. SOP mainly focuses on intersentential coherence, which is used to solve the problem of inefficient loss of next sentence prediction (NSP) in the original BERT (Yu & Reiff-Marganiec, 2022).
- In addition, the largest model, after one million steps of training, still did not overfit the training data. Therefore, the model can be larger, and ALBERT removes the dropout (dropout can be thought of as randomly removing a part of the network while making the network smaller). To speed up the training, LAMB(Layer-wise Adaptive Moments optimizer for Batching training) is used as the optimizer, and the LAMB optimizer can train a particularly large batch size (Miyato et al., 2017; Meng et al., 2021; Zheng et al., 2022).

Adversarial Training

The adversarial training can be summarized in the following maximum minimization Equation (2).

$$\min_{\theta} E_{(z,y)} \sim D\left[\max_{\Delta x \in \Omega} L\left(x + \Delta x, y; \theta\right)\right] \tag{2}$$

The inner layer (in the middle bracket) is a maximizer, where D represents the training set, x represents the input, y represents the label, θ is the model parameter, $\mathrm{L}\left(\mathrm{x}, \mathrm{y}; \theta\right)$ is the loss of a single sample, Δx is the adversarial perturbation, and Ω is the perturbation space (Liu et al., 2019; Shui et al., 2019; Zhou et al., 2019).

This equation can be understood in steps as follows:

The goal of injecting a perturbation $\Delta \mathrm{x}$ into x, Δx is to make $\mathrm{L}\left(\mathrm{x}+\Delta \mathrm{x}, \mathrm{y} ; \theta\right)$ as large as possible, that is, to make the predictions of the existing model as wrong as possible.

- Of course, Δx it is constrained, it cannot be too large, otherwise, it will not achieve the effect of "looking almost the same", so $\Delta \mathrm{x}$ has to satisfy certain constraints, the conventional constraint is $\Delta \mathrm{x} \leq \epsilon$, where ϵ is a constant.
- After constructing an adversarial sample $\mathrm{x}+\Delta \mathrm{x}$ for each sample, $\mathrm{x}+\Delta \mathrm{x}$ is used as a data pair to minimize the loss to update parameter θ (gradient descent).
- Repeat steps 1, 2, and 3 alternately.

Adversarial training serves two purposes, one is to improve the robustness of the model against malicious attacks, and the other is to improve the generalization ability of the model. In the CV(Computer Vision) task, perturbation is added directly to the original input, but for the NLP task, the "adversarial sample" constructed by this operation does not correspond to a word, so in turn, there is no way for the adversary to obtain such an adversarial sample by modifying the original input during the inference Chai & Zhu (2019)

Local Adversarial Training

In terms of the sample data itself, if the criteria are whether the samples will be misclassified when perturbations are added, the samples can be divided into simple samples, that is, those samples that are not easily perturbed and are inside the class boundary, and difficult samples that are easily perturbed and are located around the boundary or far from the correct class boundary. The global adversarial training is to add the adversarial perturbation to all the original samples directly without sample screening; the training to exclude the simple samples and add the perturbation to the difficult samples only is the local adversarial training.

Suppose X_{adv} is the adversarial sample set and ATK is the attack method for generating adversarial samples; g,l as subscripts denote the global and local methods, respectively. The set of all training samples in global adversarial training can be expressed as Equation (3).

$$X_g = X + X_{g\_adv}, \tag{3}$$

where $X_{g\_adv} = ATK_g\left(x\right)$. Suppose Hard is a difficult sample screening method, screen out the difficult samples from the original data, and then add perturbations to the difficult samples. The local adversarial sample set can be expressed as Equation (4). The set of all training samples in local adversarial training is

$$X_l = X + X_{l\_adv} \tag{4}$$

During the adversarial training, if the perturbation is added directly to all samples, a large number of simple samples are still inside the category after adding perturbations. These adversarial samples become redundant because they do not contribute to backpropagation. Therefore, adding perturbations to the filtered difficult samples to generate adversarial samples for gradient backpropagation, which can avoid generating a large number of redundant adversarial samples and the computational effort of training, can be greatly reduced.

The adversarial training method used in this paper is the Fast Gradient Sign Method (FGSM). The confusion matrix can reflect the percentage of classification errors in the samples, and the difficult samples in each category are attacked by pointing to the error class. Let the set of difficult samples of the same class be C. The sample size is S, $C=\left\{C^{(1)},C^{(2)},\cdots,C^{(s)}\right\}$, in which, each sample corresponds to the common true label l_{true}, L corresponds to the set of real labels, and the total number of label classes is N. L and C together form the training set, which is represented as Equations (5) and (6).

$$L=\left\{l_{true},\cdots,l_{true}\right\}\ Hard \tag{5}$$

$$\tilde{D}_h=Top\left(\rho,J\left(\tilde{D},\hat{\theta}\right),\tilde{D}\right)=\left\{C^{(n)},L^{(n)}\right\}_{n=1}^{N} \tag{6}$$

Let the sequence of counterattack labels corresponding to C be L_tar and use $\mathrm{conf}\left(\mathrm{L}\right)$ to denote the labels arranged by the error probability distribution of the confusion matrix, with the Equation (7) relations.

$$L_{tar}=\left\{l_{tar}\middle|l_{tar}\in L,l_{tar}\neq l_{true}\right\}=conf\left(L\right) \tag{7}$$

Let r_{tar} be the set of target attack perturbations of C, consisting of a number of unit perturbations $r_{tar}^{(s)}$, w here $\overset{o}{\alpha}$ is used to control the perturbation size; g denotes the degree; $\hat{\delta}$ denotes the set of all parameters; and let J denote the loss calculation function. The relationship is expressed as f in Equations (8) and (9).

$$r_{tar}=\left\{r_{tar}^{(s)}\right\}_{s=1}^{S} \tag{8}$$

$$r_{tar}^{(s)}=\varepsilon\frac{g^{(s)}}{g_2}, \tag{9}$$

where $g^{(s)}=\nabla_{c^{(s)}}J\left(C,L_{tar},\hat{\delta}\right)$. Let C_{tar} be the set of generated target attack adversarial samples, with L together to form the adversarial sample set $\tilde{D}_{tar}$, with $\left(C_{tar},L\right)\in\tilde{D}_{tar}$, and the adversarial sample calculation formula is as Equation (10).

$$C_{tar}=C-r_{tar}=\left\{c^{(s)}-r_{tar}^{(s)}\right\}_{s=1}^{S} \tag{10}$$

The loss function for the antagonistic sample is calculated as Equation (11).

$$J\left(\tilde{D}_{tar},\hat{\delta}\right)=\frac{1}{\left|N\right|}\sum_{(c,L)\in\tilde{D}_h}J\left(C_{tar},L,\hat{\delta}\right) \tag{11}$$

α is used to control the ratio of the original corpus to the loss value of the adversarial sample, and the total loss of the adversarial training is as Equation (12).

$$\mathrm{Loss}_{\mathrm{L\_ADV}}=\alpha\bullet\mathrm{J}\left(\tilde{\mathrm{D}},\hat{\delta}\right)+\left(1-\alpha\right)\bullet\mathrm{J}\left(\tilde{\mathrm{D}}_{\mathrm{tar}},\hat{\hat{\delta}}\right) \tag{12}$$

The optimal parameters for the adversarial training are calculated as Equation (13).

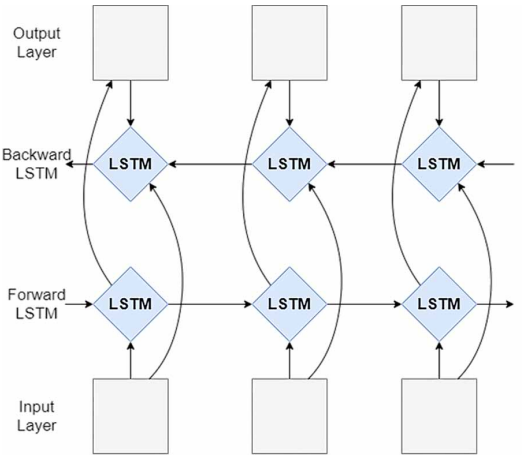
$$\hat{\delta}=\mathrm{argmin}\left\{\mathrm{Loss}_{\mathrm{L\_ADV}}\right\} \tag{13}$$

Based on the BiLSTM-CRF model, this paper combines the idea of adversarial training and difficult samples to filter the difficult samples with large loss values in the data and add target attack local perturbation. The model improves the recognition of named entities by learning the features of difficult samples near the category boundaries.

BILSTM MODULE FOR THE FUSION OF ATTENTION MECHANISM

Long short-term memory (LSTM) is a recurrent neural network (RNN), as shown in Figure 3. LSTM solves the gradient explosion and gradient disappearance problems of the basic RNN by introducing a gate mechanism and a memory unit. Briefly, an LSTM unit in the input gate module can learn to recognize an important input and store the input in a long-term state. This state is kept until it is extracted when needed, as long as it is not acted upon by the forgetting gate. Therefore, the LSTM model can be successful. The LSTM model is characterized by capturing long-term patterns. Owing to its power, it has been widely used in NLP tasks, including NER and machine translation (Pennington et al., 2014).

Figure 3. Structure of BiLSTM *Note*. **LSTM = long short-term memory.**



The unidirectional LSTM model neuronal information can only be passed from the front to the back, meaning that the input information at the current moment can only be used for the information at the previous moment. However, for the task of sequence labeling, a strong dependency occurs between the effective entities and the context. The information before the current moment and the information after it should be equally important. To merge the information from both sides of the sequence, this paper adopts a bidirectional LSTM to extract features, whose structure is shown in Figure 2. BiLSTM can use the information before the current moment and the information after and fully consider the input context information. The state can be expressed as follows.

$$\overrightarrow{\mathrm{h}_{\mathrm{i}}} = \overrightarrow{\mathrm{LSTM}}\left(\overrightarrow{\mathrm{h}_{\mathrm{i-1}}}, \mathrm{x}_{\mathrm{i}}\right) \tag{14}$$

$$\overleftarrow{\mathrm{h}_{\mathrm{i}}} = \overleftarrow{\mathrm{LSTM}}\left(\overrightarrow{\mathrm{h}_{\mathrm{i+1}}}, \mathrm{x}_{\mathrm{i}}\right) \tag{15}$$

$$\mathrm{h}_{\mathrm{i}} = \overrightarrow{\mathrm{h}_{\mathrm{i}}} \oplus \overleftarrow{\mathrm{h}_{\mathrm{i}}}, \tag{16}$$

of which, $\overrightarrow{\mathrm{h}_{\mathrm{i}}}$ and $\overleftarrow{\mathrm{h}_{\mathrm{i}}}$ are the hidden states of the LSTM before and after position i, respectively. The connection operation is denoted by $\oplus$.

The private BiLSTM layer is used to extract task-specific features, and the shared BiLSTM layer is used to learn the word boundaries shared by the task. Formally, for any sentence in task data set k, the hidden states of the shared and private BiLSTM layers can be computed as follows.

$$\mathrm{S}_{\mathrm{i}}^{\mathrm{k}} = \mathrm{BiLSTM}\left(\mathrm{x}_{\mathrm{i}}^{\mathrm{k}}, \mathrm{S}_{\mathrm{i-1}}^{\mathrm{k}}; \grave{\mathrm{e}}_{\mathrm{s}}\right) \tag{17}$$

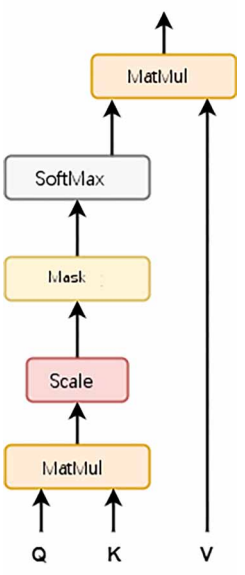
$$\mathrm{h}_{\mathrm{i}}^{\mathrm{k}} = \mathrm{BiLSTM}\left(\mathrm{x}_{\mathrm{i}}^{\mathrm{k}}, \mathrm{h}_{\mathrm{i-1}}^{\mathrm{k}}; \theta_{\mathrm{k}}\right), \tag{18}$$

of which, θ_{s} and θ_{k} are the BiLSTM parameters shared by task k and the private BiLSTM parameters, respectively.

Attention Mechanism

Attentional mechanisms originated from the study of human vision and have been widely used in applications, such as machine translation and machine vision (Choi et al., 2016)[REMOVED REF FIELD]. Due to the relevance of the input sequences, the traditional embedding representation does not consider the phase between characters, the information in the input sequences is not fully utilized. Therefore, an attention mechanism is introduced to deeply extract lexical features and semantic information, so that the characters related to entities in the Chinese data set are automatically attended to, useless information is ignored, and local features of long text sequences are taken into account. The structure is shown in Figure 4.

Figure 4. Attention Mechanism Structure *Note.* **Q= Query;K=Key;V=Value.**



Let the bidirectional LSTM output vector be $\mathrm{H_i} = \left[\overrightarrow{\mathrm{h_i}} \oplus \overleftarrow{\mathrm{h_i}}\right]$. The importance of the first, or the first character in the sentence, is quantified as the energy function, the location of the bias component.

$$\mathrm{e_i} = \tanh\left(\mathrm{w}^{\bar{\mathrm{T}}}\mathrm{H_i} + \mathrm{b}\right) \tag{19}$$

The result of normalizing is denoted as

$$\alpha_{\mathrm{i}} = \frac{\exp\left(\mathrm{e_i}\right)}{\sum_{\mathrm{i}} \exp\left(\mathrm{e_i}\right)}. \tag{20}$$

The attention weights are calculated based on the dynamic scale as

$$\mathrm{H_i^{'}} = \mathrm{H_i} \cdot \alpha_{\mathrm{i}}. \tag{21}$$

The attention weight assignment method is used to change the probability matrix of bidirectional LSTM output, which can take into account more local features and improve sequence annotation results for CRF layers.

CRF Module

The CRF belongs to the probabilistic undirected graph model in the probabilistic graph model. That is, a joint probability distribution $P\left(Y\right)$ is represented by the undirected graph $G = \left(V, E\right)$ represented by the graph G. The set of nodes of V denotes Y of a series of random variables. The set E of edges denotes the dependence between the random variables. If the joint probability

distribution $P\left(Y\right)$ satisfies pairwise, local, or global Markovness (which is equivalent to the fact that each random variable in the graph, and the random variables adjacent to it, are conditionally independent of each other given that the points in the graph are not adjacent to them), then this joint probability is said to be a probabilistic undirected graph model, that is, a CRF (Choi et al., 2016; Zhou et al., 2019).

The greatest characteristic of the probabilistic undirected graph model is that the joint probability is easy to factorize. Additionally, the joint probability can be easily decomposed into the form of probability multiplication by the potential function on the largest group, facilitating the calculation of probabilities (Ma et al., 2022).

The CRF model is used to solve the problem of predicting the conditional distribution of another set of output random variables given a set of input random variables. The model is predicated on the assumption that the random variables constitute a Markov random field. The linear chain conditional random field was proposed by Lafferty in 2001. At present, the linear chain CRF model is the most classical approach to solve the sequence labeling problem. Specifically, if the observation sequence is Equation (22), the one-to-one corresponding labeled sequence is Equation (23). The CRF model is to construct the conditional probability between the two $P\left(Y\middle|X\right)$.

$$X=\left\{x_1,x_2,\cdots,x_n\right\} \tag{22}$$

$$Y=\left\{y_1,y_2,\cdots,y_n\right\} \tag{23}$$

The CRF labeling process can be formalized as follows.

$$o_i=w_s\text{"}+b_s \tag{24}$$

$$s\left(x,y\right)=\sum_{i=1}^{N}\left(o_{i,y_i}+T_{y_{i-1},y_i}\right) \tag{25}$$

$$\overline{y}=arg\max_{v\in Y_v}s\left(X,Y\right), \tag{26}$$

where w_s and b_s are trainable parameters, o_{i,y_i} represents the character x_i of the first y_i score of the first tag, T is a transformed score matrix that defines the scores of two consecutive tags, and Y_x denotes the sequence of all candidate tags for a given sentence x. In the decoding process, we use the Viterbi algorithm to obtain the predicted tag sequences. In the training, we utilize the negative log-likelihood objective as the loss function. The probability of the ground truth label sequence is calculated as

$$P\left(\hat{y}\middle| X\right)=\frac{e^{s\left(X,\hat{y}\right)}}{\sum_{\tilde{y}\in Y_X} e^{s\left(x,\hat{y}\right)}}, \tag{27}$$

where Y_X denotes the set of all possible labels, the numerator s function denotes the score of the correct label, and the denominator s function denotes the sum of the scores of each possible label. In the training process of the CRF model, the loss function is defined as:

$$L_{ner}=-\log P\left(\hat{y}\middle| x\right), \tag{28}$$

The value of the loss function is calculated, and the network parameters are continuously updated until the end of the iteration. The CRF is a log-linear model defined on time-series data with the ability to express long distance dependence and intersection features, which can better solve problems such as labeling bias. Many studies show that the model is useful for NER techniques.

EXPERIMENT

Experimental Setup

The open-source text classification data set THUCTC(Tsinghua Chinese Text Classification) from Tsinghua University and The People's Daily 2004 news corpus is used as the data set. In this paper, the data sets are fielded and textually annotated by manually using three entity types (person, place, institution names). Table 2 shows the detailed number of data sets used in this paper.

Table 2. Data Set

Data set	Training set	Validation set
People'sDaily	20864	2318
THUCTC	10748	1343

Note. THUCTC = tsinghua chinese text classification.

Three algorithms were chosen for the comparison:

(1) Comparison experiments of different embedding methods

The experiment uses the ALBERT, BERT, and Word2Vec embedding methods to obtain the vector representation of the input after BILSTM-CRF, BILSTM-Attention-CRF for Chinese NER task experiments. This comparative experiment used The People's Daily 2004 news data set, and the perturbation factor e is 0.5. The experimental results are shown in Table 5.

(2) Comparison experiments of different Chinese NER models

This experiment adopts the Word2Vec + Adversarial training + BiLSTM + Attention + CRF, and the model AAA(ALBERT+ Adversarial + Attention + BiLSTM + CRF (the model proposed in this paper, ALBERT + Adversarial training + BILSTM + Attention + CRF) for comparison in

experiments of Chinese NER tasks. This comparative experiment used The People's Daily 2004 news and THUCTC data sets, and the perturbation factor e is 0.5. The experimental results are shown in Table 6.

(3) Comparison experiments of different perturbation factors

For the AAA + BiLSTM + CRF and Word2Vec + Adversarial training + BiLSTM + Attention + CRF model, different perturbation factors were set for the experiments to analyze the effects of different perturbation factors on the experiments. The data set used was The People's Daily. The experimental results are shown in Table 7.

Evaluation Metric

In this paper, the experimental evaluation metrics are Precision (P), Recall (R), and F1 values, which are used to evaluate the performance of the Chinese NER model, and a predicted tagged entity is considered correct when and only when it matches the real entity:

$$Precision = \frac{Identify\,the\,correct\,number\,of\,entities}{Number\,of\,entities\,identified} \times 100\% \tag{29}$$

$$Recall = \frac{Identify\,the\,correct\,number\,of\,entities}{Number\,of\,entities\,in\,the\,sample\,set} \times 100\% \tag{30}$$

$$F_1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \times 100\% \tag{31}$$

Experimental Environment

This experiment is based on the TensorFlow deep learning framework design, using the Python language implementation. The experimental running platform is pycharm2021.2.3 (64-bit), the video memory is 8 GB, and the GPU is NVidia GTX 1080.

Model Parameter Settings

The Gaussian error linear unit algorithm is used as the activation function, and the random deactivation rate (dropout) function is used to reduce the effect of overfitting.

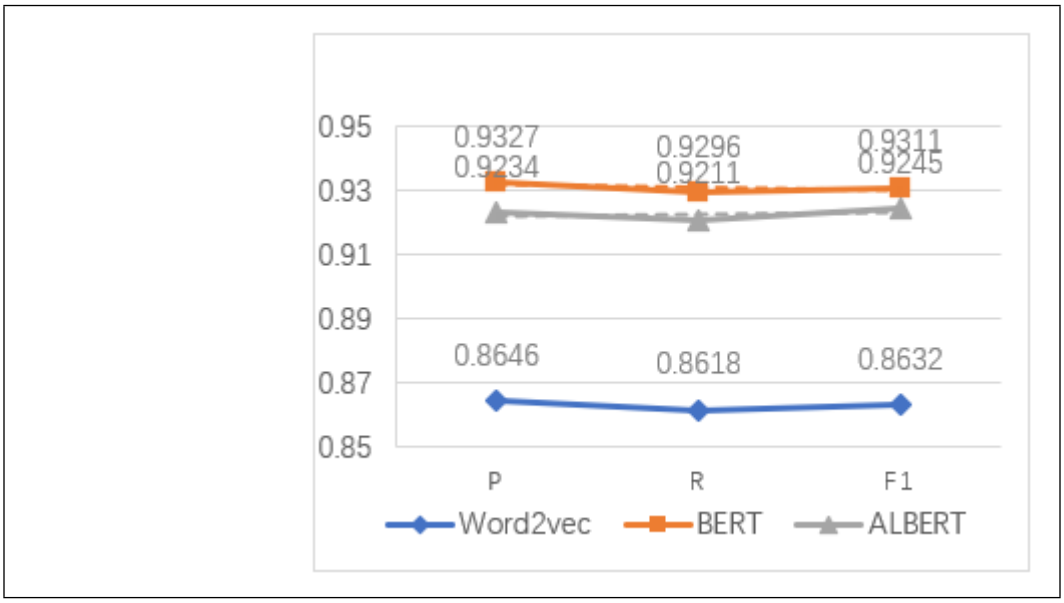
The main model parameters are set as follows:

The length of the character vector is 200, the size of the ALBERT hidden layer is 768, the ALBERT learning rate is 1e-4, the dropout value is 0.1, the training round epoch is 10, and the batch size is set to 16.0.

RESULTS

Analysis of the experimental results leads to the following conclusions.

Table 3. Results of the BILSTM-CRF Experiments Under Different Embedding Methods



1) As shown in Table 3, on the baseline model BiLSTM + CRF, the F1 of ALBERT showed an improvement of 6.13 over the Word2Vec method and a slight decrease of 0.66 over the BERT method, BERT performed best.

Table 4. Comparison of the Results of Model Experiment I

Embedding mode	Model	P	R	F1
Word-2vec	BiLSTM + CRF	0.8646	0.8618	0.8632
BERT		0.9327	0.9296	0.9311
ALBERT		0.9234	0.9211	0.9245
BERT	BiLSTM+MHA+CRF	0.9378	0.9342	0.9360
Word-2vec	BiLSTM + Attention + CRF	0.8831	0.8769	0.8789
ALBERT		0.9362	0.9285	0.9344

Note. ALBERT =a lite bert ; BERT = bidirectional encoder representations from transformers; BILSTM = bidirectional long short-term memory; CRF = conditional random field; MHA = multiheaded attention mechanism.

2) As shown in Table4, adding the attention layer after the feature extraction layer can take into account the local features of the sequence text, and the performance of the NER algorithm has a small improvement. Compared with the baseline model BILSTM-CRF, ALBERT improved the F1 value by 5.55 over the Word2Vec embedding method. Compared to the model ALBERT+BILSTM+CRF, the addition of the attention mechanism improved the F1 value by 0.99. The best performer was the BERT + BiLSTM + MHA (multiheaded attention mechanism)

+ CRF model (Ma et al., 2022), which achieved an F1 value of 0.9360. This was slightly higher than the ALBERT+BILSTM+Attention+CRF model, and it may be because the multiheaded attention mechanism performs better on this data set.

Table 5. Comparison of the Results of Model Experiment II

Model	Data set	P	R	F1
Word-2vec+Adversarial Training + BiLSTM +Attention + CRF	People's Daily	0.9257	0.8916	0.8947
	THUCTC	0.9411	0.8593	0.8989
AAA+ BiLSTM+ CRF (methods proposed in this paper)	People's Daily	0.9471	0.9242	0.9364
	THUCTC	0.9518	0.9319	0.9435

Note. BILSTM = bidirectional long short-term memory; CRF = conditional random field; THUCTC = tsinghua chinese text classification. .

3) As shown in Tables 4 and 5SSS, the experimental data show that the method with the addition of adversarial training in the Word2Vec embedding approach improves the F1 value by 3.15 over the baseline model BILSTM-CRF. In the ALBERT embedding method, the F1 value of the model, proposed in this paper is significantly higher, by 1.19 over ALBERT+BiLSTM+CRF. However, in Table 5, under the same conditions, the F1 value of People's Daily is lower when doing the NER task on the THUCTC. The reason for this result is that the sample size in People's Daily is larger, providing more sufficient features for model learning.

Therefore, adversarial training cannot play an optimal role in such large sample data. By contrast, the relatively small sample size of the THUCTC with a relatively small number of samples subtly expands the corpus data by adding only slightly perturbed adversarial samples distributed around the original samples. This addition has a positive significance for the model to fully learn the sample features.

Table 6. Results of Experiment III

Model	Perturbation factor	P	R	F1
Word-2vec + Adversarial Training + BiLSTM + Attention + CRF	0.3	0.9279	0.8489	0.8864
	0.5	0.9257	0.8916	0.8947
	0.8	0.9321	0.8522	0.8878
AAA + BiLSTM (methods proposed in this paper)	0.3	0.9288	0.8512	0.8894
	0.5	0.9471	0.9242	0.9364
	0.8	0.9428	0.8873	0.9112

Note. AAA =*a lite bert*+ adversarial+attention ; BILSTM = bidirectional long short-term memory; CRF = conditional random field.

4) The experimental data in Table 6 show that different perturbation factors affect the experimental results differently (and not the larger the perturbation factor, the better the experimental results). Among the three perturbation factors set, 0.5 has the best effect.

CONCLUSION

To improve the accuracy rate of named entity recognition on Chinese data sets, this paper proposes a Chinese NER method that incorporates ALBERT adversarial training and attention mechanism: AAA+BiLSTM+CRF.

For this experiment, two Chinese data sets, People's Daily 2004 and Tsinghua University open-source text classification, are collected and collated. Inspired by transfer learning, the method first introduces the ALBERT pretrained language model, and the obtained word vector can enhance the dependency relationship between any two words. The idea of difficult sample screening is adopted for the word vector to screen out the difficult samples containing a large number of boundary samples that have a critical effect on the model performance. Using the property that the boundary samples are easily perturbed, we combine the target attack method based on the error probability distribution of the confusion matrix to generate the samples used for adversarial training. After the BiLSTM coding, the task's shared and private features are obtained, filtered by the attention layer. The best entity annotation is obtained by CRF.

Through experiments, the proposed model is proven to obtain good experimental performance in Chinese NER and has certain effectiveness.

Future research efforts will continue in the area of NER around the accuracy rate of related algorithms based on migration learning and adversarial training on domain-specific data sets, for example, in fields such as architecture.

The authors of this publication declare there is no conflict of interest.

This work is supported by the Anhui Provincial Natural Science Foundation 2008085MF218, the Provincial Natural Science Research project of universities in Anhui Province, KJ 2021A0623, the Open Fund of Key Laboratory of Flight Techniques and Flight Safety, CIVIL AVIATION FL IGHT UNIVERSITY OF CHINA (No. FZ2021KF10), the university launched the talent introduction scientific research launch project 2019QDZ38, the Academic Grant Project for Top Discipline Talents in Anhui Province (gxbjZD26).

REFERENCES

Agrawal, A., Tripathi, S., Vardhan, M., Sihag, V. K., Choudhary, G., & Dragoni, N. (2022). BERT-based transfer-learning approach for nested named-entity recognition using joint labeling. *Applied Sciences (Basel, Switzerland)*, *12*(3), 976. doi:10.3390/app12030976

Alzantot, M. F., Sharma, Y., Elgohary, A., Ho, B., Srivastava, M. B., & Chang, K. (2018). Generating natural language adversarial examples. *EMNLP*. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (pp. 2890–2896). Association for Computational Linguistics. doi:10.18653/v1/D18-1316

Bikel, D. M., Miller, S., Schwartz, R., & Weischedel, R. (1998). Nymble: A high-performance learning name-finder. In *Fifth Conference on Applied Natural Language Processing* (pp. 194–201). Association for Computational Linguistics. doi:10.3115/974557.974586

Chai, M., & Zhu, Y. (2019). Research and application progress of generative adversarial networks. *Computer Engineering.*, (9), 222–234. doi:10.19678/j.issn.1000-3428.0051964

Choi, E., Bahadori, M. T., Sun, J., Kulas, J. A., Schuetz, A., & Stewart, W. F. (2016). *RETAIN: An interpretable predictive model for healthcare using reverse time attention mechanism*. Neural Information Processing Systems NIPS. doi:10.48550/arXiv.1608.05745

Dong, Z., Shao, R., Chen, Y., & Zhai, W. (2021). Named entity recognition in food field based on BERT and confrontation training. *2021 33rd Chinese Control and Decision Conference (CCDC)*, 2219–2226. doi:10.1109/CCDC52312.2021.9601522

Gong, Z., Wang, W., Li, B., Song, D. X., & Ku, W. (2018). Adversarial Texts with Gradient Methods. *Computation and Language*. Advance online publication. doi:10.48550/arXiv.1801.07175

Haoliang, X. U., Yanqun, L. I., Yunqi, H. E., & Qian, L. (2019). Research on Chinese nested named entity relation extraction. *Beijing Da Xue Xue Bao. Zi Ran Ke Xue Bao*, *55*(1), 8–14. doi:10.13209/j.0479-8023.2018.056

He, Y., Du, F., Shi, Y., & Song, L. (2021). A review of named entity recognition based on deep learning. *Computer Engineering and Application*, (11), 21–36.

Huang, Z., Xu, W., & Yu, K. (2015). *Bidirectional LSTM-CRF Models for Sequence Tagging*. ArXiv, abs/1508.01991.

Hui, W. U., Li, L. V., & Bi-Hui, Y. U. (2019). Chinese named entity recognition based on transfer learning and bilstm-CRF. *Journal of Chinese Computer Systems*, *40*(6), 1142–1147.

Isozaki, H., & Kazawa, H. (2002). Efficient support vector classifiers for named entity recognition. *COLING '02: Proceedings of the 19th International Conference on Computational Linguistics*, 1, 1–7. doi:10.3115/1072228.1072282

Lafferty, J. D., McCallum, A., & Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML '01: Proceedings of the Eighteenth International Conference on Machine Learning*, 282–289.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2020). *ALBERT: A Lite BERT for Self-supervised Learning of Language Representations*. ArXiv, abs/1909.11942.

Li, J., Ji, S., Du, T., Li, B., & Wang, T. (2019). *TextBugger: Generating Adversarial Text Against Real-world Applications*. ArXiv, abs/1812.05271.

Li, N., Guan, H., Yang, P., & Dong, W. (2020). A Chinese named entity recognition method based on BERT-IDCNN-CRF. *Journal of Shandong University*, (1), 102–109.

Liu, Q., Li, Y., Duan, H., Liu, Y., & Qin, Z. (2016). Knowledge graph construction techniques. *Journal of Computer Research and Development*. 10.7544/ISSN1000-1239.2016.20148228

Liu, J. (2021). *Research on Migration Learning Algorithm Based on Countermeasure Network* [Master's Thesis]. University of Electronic Science and Technology of China. https://kns.cnki.net/KCMS/detail/detail.aspx?dbname=CMFD202201&filename=1021745101.nh

Liu, J., Cheng, J., Wang, Z., Lou, C., Shen, C., & Sheng, V. S. (2022). A survey of deep learning for named entity recognition in Chinese social media. In Artificial Intelligence and Security. ICAIS 2022. Lecture Notes in Computer Science (vol 13338). Springer. ICAIS. doi:10.1007/978-3-031-06794-5_46

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. ArXiv, abs/1907.11692.

Ma, L., Li, T., Liu, A., & Qin, J. (2022). Research on named entity recognition of small dataset based on transfer learning. *Journal of Huazhong University of Science and Technology*, (2), 118–123. doi:10.13245/j.hust.220218

Mandle, A. K., Sahu, S. P., & Gupta, G. P. (2022). CNN-based deep learning technique for the brain tumor identification and classification in MRI images. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–20. doi:10.4018/IJSSCI.304438

Meng, L., Yanling, L., & Min, L. (2021). A review of transfer learning in named entity recognition *Computer Science and Exploration,* (2), 206–218. 10.3778/j.issn.1673-9418.2003049

MikolovT.ChenK.CorradoG. S.DeanJ. (2013). Efficient Estimation of Word Representations in Vector Space. *ICLR*. arXiv:1301.3781v3

MiyatoT.DaiA. M.GoodfellowI. J. (2017*). Adversarial Training Methods for Semi-Supervised Text Classification.* arXiv:1605.07725v4

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. *EMNLP*. https://nlp.stanford.edu/projects/glove/

Shui, L., Liu, W., & Feng, Z. (2019). Automatic image annotation based on generative adversarial network. *Jisuanji Yingyong*, *39*(7), 2129–2133.

Wang, C., Chen, W., & Xu, B. (2017). Named entity recognition with gated convolutional neural networks. In M. Sun, X. Wang, B. Chang, & D. Xiong (Eds.), Lecture Notes in Computer Science: Vol. 10565. *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data. NLP-NABD CCL 2017 2017*. Springer. doi:10.1007/978-3-319-69005-6_10

Wang, H., Shen, Q., & Xian, Y. (2017). Chinese named entity recognition based on integrated transfer learning. *Minicomputer System*, (2), 346–351.

Xu, G., Meng, Y., Zhou, X., Yu, Z., Wu, X., & Zhang, L. (2019). Chinese event detection based on multi-feature fusion and BiLSTM. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 134992–135004. doi:10.1109/ACCESS.2019.2941653

Yang, P., & Dong, W. (2020). Chinese named entity recognition method based on BERT embedding. *Computer Engineering,* (4), 40–45+52. 10.19678/j.issn.1000-3428.0054272

YangZ.DaiZ.YangY.CarbonellJ. G.SalakhutdinovR.LeQ. V. (2019). *XLNet: Generalized Autoregressive Pretraining for Language Understanding*. NeurIPS. arXiv:1906.08237v2

Yu, H. Q., & Reiff-Marganiec, S. (2022). Learning disease causality knowledge from the web of health data. *International Journal on Semantic Web and Information Systems*, *18*(1), 1–19. doi:10.4018/IJSWIS.297145

Zhang, J., Bi, Z., Wang, J., & Wu, X. (2019). Design of Chinese domain named entity recognition framework based on BLSTM CRF. *Jisuan Jishu Yu Zidonghua*, *38*(3), 5.

Zhang, Y., & Yang, J. (2018). Chinese NER using lattice LSTM. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics* (*Volume 1: Long Papers*). Association for Computational Linguistics. https://aclanthology.org/P18-1144

Zheng, C., Wang, X., Wang, T., Deng, Y., & Yin, T. (2022). A multi label medical text classification method based on the ALBERT TextCNN model. *Journal of Shandong University*. *Science Edition*, (4), 21–29.

Zhong, Q., & Tang, Y. (2020). An attention-based BILSTM-CRF for Chinese named entity recognition. *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA),* 550–555. doi:10.1109/ICCCBDA49378.2020.9095727

Zhou, J. T., Zhang, H., Jin, D., Zhu, H., Fang, M., Goh, R., & Kwok, K. (2019). Dual adversarial neural transfer for low-resource named entity recognition. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.* https://aclanthology.org/P19-1336