ISSN 2616-5775 Vol. 6, Issue 6: 8-15, DOI: 10.25236/AJCIS.2023.060602

Chinese Named Entity Recognition Based on Multi-Feature Fusion FLAT Model in the Medical Field

Ning Wang, Lin Ni*

University of Science and Technology of China, Hefei, 230027, Anhui, China
nilin@ustc.edu.cn
**Corresponding Author*

Abstract: *The complexity of syntax and the specialized nature of Chinese electronic medical record data make it challenging to accurately identify medical entities using named entity recognition models. In order to precisely extract complex medical vocabulary from electronic medical records, this paper proposes a multi-feature-based named entity recognition model that addresses the issue of insufficient internal feature extraction in the FLAT model. Firstly, the radical and pinyin features of Chinese characters are extracted for enrichment and perfection of their semantic information. These features are then combined with the word embeddings extracted by the FLAT-lattice method. Finally, the fused features and position encoding are input to the Transformer encoder for encoding, followed by decoding using the CRF method. Experimental results demonstrate that the proposed model outperforms many existing algorithms on the CCL2021 dataset, with an F1 score of up to 91.75%.*

Keywords: *Chinese Electronic Medical Records; Named Entity Recognition; FLAT-Lattice*

1. Introduction

With the rapid development and application of hospital information systems, large-scale electronic medical record (EMR) data generated during patient visits and treatment have been accumulated in medical institutions, including medical texts, medical charts, medical images, and other types of data. Among them, the unstructured EMR text data are the most important part, such as chief complaints, diagnostic results, admission/discharge records, and treatment processes, which contain a wealth of valuable medical knowledge and health information. Named entity recognition (NER) for EMRs, which identifies medically-related entity names from unstructured texts and categorizes them into predefined categories such as diseases, treatments, symptoms, and drugs, is a crucial step in EMR data mining and information extraction. NER not only serves as a solid foundation for natural language processing (NLP) related tasks such as information retrieval, information extraction, and question-answering systems but also plays a huge role in various applications of EMRs, such as comorbidity analysis, adverse drug event detection, and drug interaction analysis.

When it comes to English named entity recognition tasks, the presence of spaces between words serves as a natural delimiter, facilitating the identification of entity boundaries. However, there is no similar delimiter setup among Chinese characters, which poses a significant challenge for Chinese named entity recognition. The absence of whitespace and the lack of clear boundary markers lead to great difficulties in accurately identifying entity boundaries. As a result, Chinese named entity recognition requires more complex techniques that are capable for incorporating contextual, syntactic, and semantic information present in sentences.

In the realm of Chinese named entity recognition models, there exists a category of models that rely on word information for entity recognition and classification by typically using word segmentation to determine entity boundaries and subsequently adopting a named entity recognition model to perform sequence annotation[1-3]. Therefore, it is evident that the accuracy of subwords directly affects the determination of entity boundaries, thus influencing the precision of entity recognition[4]. On the other hand, character-based feature recognition models can directly take a single Chinese character as input without requiring word segmentation[5, 6], effectively circumventing the challenges brought about by word segmentation errors, which largely outperform word-based models[7, 8]. Nonetheless, the inclusion of lexical-level information without being leveraged by character-based feature recognition

ISSN 2616-5775 Vol. 6, Issue 6: 8-15, DOI: 10.25236/AJCIS.2023.060602

models can indeed enhance the accuracy of feature boundary determination and augment the semantic representation of sentences. To address the aforementioned challenges, Lite-LSTM [9] introduces a lexicon and incorporates relevant lexical information into the character-level LSTM model to enhance the model's semantic representation and features like accuracy of entity boundary discrimination. This approach effectively mitigates the problem of error propagation resulting from word segmentation errors, thereby improving the accuracy of Chinese named entity recognition.

The proposed Lattice-LSTM model is a critical advancement in incorporating lexical information into character-based Chinese named entity recognition models, but it still suffers from several limitations. Lattice-LSTM introduces supplementary edges between characters to integrate relevant word information. The inconsistency in the number of potential words corresponding to each character causes obstacles in achieving batch parallelization during model learning and inference. Furthermore, the lattice structure of the model enables each character to capture information solely about the word that concludes with it, while ignoring the word in its intermediate section, resulting in information loss.

The FLAT model[10] provides a promising solution to the aforementioned issues. It concatenates all matched words into a flat structured word sequence behind the input characters in the input layer. With the proposal of a position encoding technique to incorporate relative position information into the model, the traditional LSTM model in the encoding layer is replaced with the encoding layer of the Transformer to encode information about the input words. The self-attention mechanism of the Transformer facilitates interaction between input characters and their corresponding potential words, thus eliminating information loss and increasing parallel computing efficiency. Despite the FLAT model's introduction of word vectors through word2vec to leverage semantic information between characters, it neglects the rich semantic information inherent in Chinese characters. As advanced pictograms, Chinese characters contain a wealth of information, such as strokes, radicals, and phonetic intonation, which are intertwined and complemented to form the semantic information of Chinese characters. Studies have shown that Chinese characters with similar features, structures, and pinyin tend to present similar semantic expressions[11].

In response to inadequate internal semantic representation of Chinese characters, various methods have been proposed to extract character embeddings and incorporate additional features to strengthen their semantic representation. For instance, Chen[12] employed the GRU-GatedConv network to extract character embedding features, which were then fused with radical and lexical embeddings before being fed to the character recognition algorithm. Similarly, Yang[13] realized named entity recognition through integrating the extracted features with radical features with the assistance of a character-based BiLSTM-CRF model. In spite of the effectiveness confirmed, these models suffer from limited parallel computational capabilities due to their reliance on BiLSTM-CRF models. In contrast, pinyin, an important feature for expressing the semantics of Chinese characters, has been extensively studied in the field of pre-entrant word embedding. By integrating pinyin features into a word embedding model, Zhu[11] demonstrated that the integration of pinyin features could improve the quality of word embeddings through lexical similarity and text classification experiments. Similarly, Zhang[14] proposed the ssp2vec word embedding model, which incorporates pinyin features and Chinese character radicals to perform systematic learning and training of Chinese word embeddings. This model was proved to be effective through basic text classification and named entity recognition experiments. Generally, these models provide a more comprehensive semantic representation of Chinese characters on the basis of integration of pinyin features and character embeddings.

Based on the above studies, a FLAT model that integrates pinyin and radical features is proposed in this paper. The model employs a one-dimensional convolutional neural network to extract radical and pinyin features of Chinese characters, which are then combined with the characters and potential medical phrase embeddings generated by the FLAT model. Afterwards, the resulting coupled embeddings and positional codes are fed into the encoding layer of the Transformer for further encoding, and the prediction sequence is obtained through decoding with a conditional random field. Notably, this model solves the issue of insufficient internal feature extraction of Chinese characters by incorporating pinyin and radical features into the embedding layer, and also improves boundary features by adding lexical information. Moreover, the replacement of the LSTM model in the encoding layer by a fully connected self-attentive structure of the coding layer of the Transformer is feasible and effective, which allows for parallel computation and capture of long-range dependencies.

ISSN 2616-5775 Vol. 6, Issue 6: 8-15, DOI: 10.25236/AJCIS.2023.060602

2. Method

As mentioned before, a FLAT model with a three-layer structure that integrates pinyin features and radical features is put forward to enhance the performance of Chinese named entity recognition models. Specifically, an embedding layer, an encoding layer and a decoding layer constitute the whole architecture. The embedding layer combines pinyin, radical and word features to reinforce the semantic representation of Chinese characters. The encoding layer captures long-range dependencies among these characters and potential words with the help of the Transformer encoder. The encoded character vectors are extracted and decoded using the CRF in the decoding layer, yielding the predicted label sequence corresponding to the characters. Overall, the proposed FLAT model offers a novel approach to boosting the accuracy of Chinese named entity recognition models.

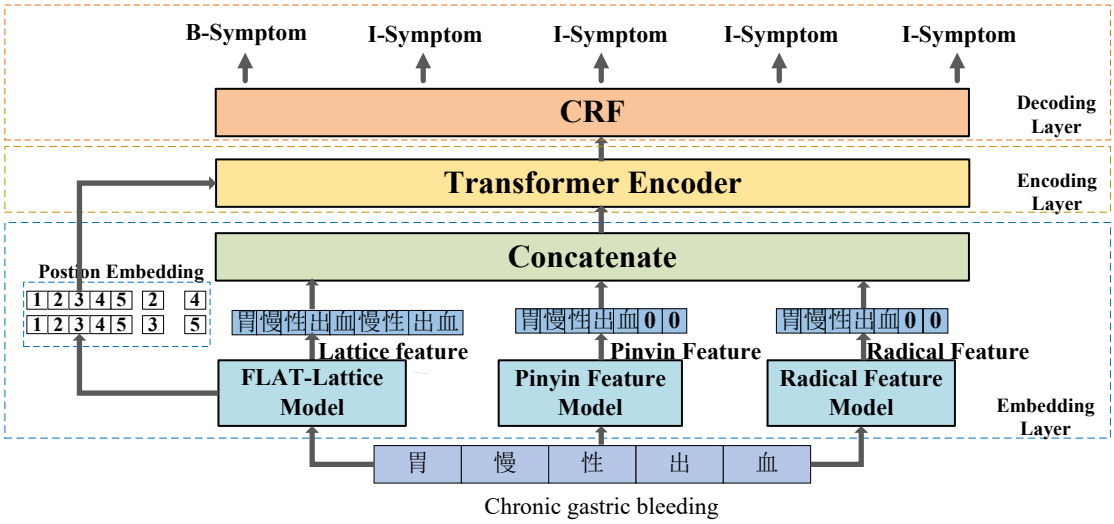

Figure 1: Overall Structure of Multi-feature Fusion FLAT Model

2.1. Embedding Layer

The embedding layer comprises three key components: (1) word and position integration generated by the FLAT-Lattice module; (2) pinyin integration obtained from the pinyin feature module; (3) radical integration obtained from the radical feature module. The subsequent sections provide a comprehensive overview of each of these modules.

2.1.1. FLAT-Lattice Module

The FLAT model incorporates the FLAT-Lattice module to match each character in a sentence to its corresponding group of potential words and retrieve the corresponding pre-trained word vector from a word vector table. Additionally, relative positions of Chinese characters and potential words are encoded. For example, given the statement "慢性胃出血 (chronic gastric bleeding)", the FLAT-Lattice module offers the mapped word sequences and positions in Fig.2 through mapping these characters to the potential words of "慢性 (chronic)" and "出血 (bleeding)". The entire process is illustrated in detail below in a scholarly style.

Figure 2: FLAT-Lattice with Character and Word Position Inputs

Assuming an input sequence of length n is denoted as $R = \{r_1, r_2, \ldots, r_n\}$, the FLAT-Lattice module performs lexical matching for each character in the sequence. Assuming that m words are mapped, the word sequence is searched in an array of pre-trained word vectors to obtain the corresponding word embedding sequence. The specific description is as follows.

$$LS = \{ls_1, ls_2, \ldots, ls_n, ls_{n+1}, \ldots, l_{n+m}\} \quad (1)$$

ISSN 2616-5775 Vol. 6, Issue 6: 8-15, DOI: 10.25236/AJCIS.2023.060602

The process of positional encoding is explained as below. The FLAT-Lattice model utilizes relative position encoding to represent the boundary relationships between entities based on the positions of characters and words in the word sequence. The process of relative position encoding is carried out as follows. First, both starting and ending positions of each character or word in the word sequence are determined, which represent their respective positions. Next, the four relative positions of both starting and ending positions of each pair of characters or words are calculated, and the relative position matrix of characters or words is achieved using Equations (2) to (5). A detailed description of this process is presented below.

$$d_{ij}^{(hh)} = head[i] - head[j] \tag{2}$$

$$d_{ij}^{(ht)} = head[i] - tail[j] \tag{3}$$

$$d_{ij}^{(th)} = tail[i] - head[j] \tag{4}$$

$$d_{ij}^{(tt)} = tail[j] - tail[j] \tag{5}$$

Where $head[j]$ stands for the head position of a character or word j, and $tail[j]$ indicates the tail position of the character or word j. Then, by using trigonometric functions, the relative position matrices are encoded based on Equations (6) and (7) to obtain the positional vector P_d.

$$P_d^{(2k)} = sin(d/10000^{2k/d_{model}}) \tag{6}$$

$$P_d^{(2k+1)} = cos(d/10000^{2k/d_{model}}) \tag{7}$$

Where d represents the relative distance between the input characters or words, k represents the index of the dimension of the positional encoding, and d_{model} refers to the vector dimension of the positional encoding. Finally, the four relative position vectors are concatenated and transformed non-linearly to acquire the relative position encoding vector R_{ij}, as shown in Equation (8).

$$R_{ij} = ReLU(W_p(P_{d_{ij}^{(hh)}} \oplus P_{d_{ij}^{(th)}} \oplus P_{d_{ij}^{(ht)}} \oplus P_{d_{ij}^{(tt)}})) \tag{8}$$

Where $ReLU$ denotes the activation function, $W_P$is the learnable parameter, and $\oplus$ means the concatenation operator, which jointly concatenate the four position vectors.

2.1.2. Pinyin Feature Module

The pinyin feature extraction module adopts a one-dimensional convolutional neural network to extract pinyin features. Fig.3 provides an illustration of the feature extraction process for the input statement "慢性胃出血(chronic stomach bleeding)". The procedures are outlined as follows: (1) The pinyin sequence for each character in the sentence is acquired by the open-source toolkit. Pinyin refers to a sequence of Romanian characters with five tones. For instance, the pinyin sequence for "慢 (slow)" is "man 4". (2) In the expression of "慢性胃出血 (chronic stomach bleeding)", the character with the highest number of pinyin sequences is "性 (nature)", which has five pinyin numbers. Therefore, the pinyin number of each character in the sentence is set to five, and any character with less than five pinyin numbers is filled with "<pad>". (3) An initialized vector representation of pinyin is obtained from the vector dictionary of pinyin components. (4) These pinyin features are extracted using a one-dimensional convolutional network, leading to corresponding pinyin feature vectors of the characters.

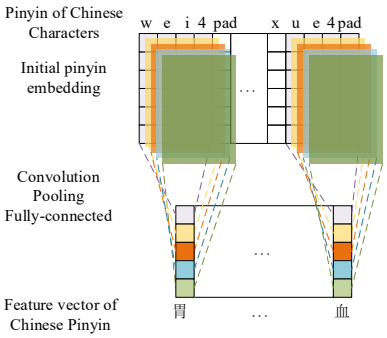

Figure 3: Process of extracting pinyin features of Chinese characters.

ISSN 2616-5775 Vol. 6, Issue 6: 8-15, DOI: 10.25236/AJCIS.2023.060602

If the input character sequence $R = \{r_1, r_2, \dots, r_n\}$ has a length of n, then the pinyin feature sequence obtained by the pinyin feature extraction module can be represented as:

$$PF = \{pf_1, pf_2, \dots, pf_n\} \tag{11}$$

2.1.3. Radical Feature Module

The process of radical extraction for the entire input statement is interpreted as follows. Firstly, each character in the input statement is disassembled using glyph analysis tools to obtain corresponding radicals, with the maximum number of radicals counted. Characters with insufficient radicals are then filled in with "<pad>" according to the maximum number of radicals. Next, with radical integration of each character conducted under the initial vector table, the radical feature sequence integration is achieved by extracting radical features of these characters using the convolutional network. Assuming the length of the input sequence is n, i.e., $R = \{r_1, r_2, \dots, r_n\}$, after extraction by the radical feature extraction module, the radical feature sequence of the sentence is expressed as follows.

$$RF = \{rf_1, rf_2, \dots, rf_n\} \tag{10}$$

The input word sequence $R = \{r_1, r_2, \dots, r_n\}$ is processed by the three aforementioned modules to obtain the word embedding LS, structural feature embedding RF, and phonetic feature embedding PF. As the last two modules only extract character features instead of lexical features, "<pad>" is used to fill RF and PF to match the length of LS. The resulting filled vectors are then concatenated with the lexical vector and utilized as the output of the integration layer, resulting in the integrated vector.

$$E = LS \oplus RF \oplus PF \tag{12}$$

2.2. Encoding Layer

The encoding layer adopts one encoder of the Transformer model, as displayed in **Fig.4**.

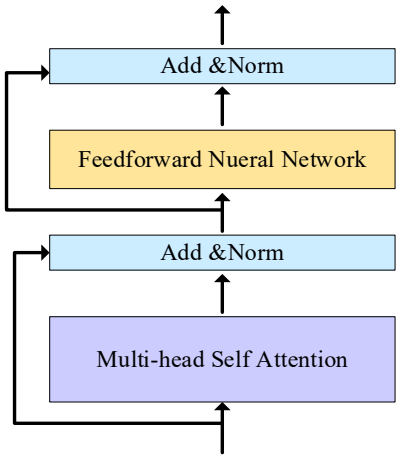

Figure 4: Transformer Encoder

First, multi-feature embeddings E and relative positional encoding vectors R_{ij} are encoded using the multi-head self-attention mechanism, as expressed in Equations (13) to (15):

$$[Q, K, V] = E[W_q, W_k, W_v] \tag{13}$$

$$S_{ij} = (Q_i + u)K_j + (Q_i + v)^T R_{ij} W_R \tag{14}$$

$$A = E + Softmax(S)V \tag{15}$$

In the equation, $W_q, W_k, W_v,$ and W_R are all learnable parameters, which represent the query mapping matrix, key mapping matrix, value mapping matrix, and position mapping matrix, respectively.

The encoded output of the multi-head self-attention is fed into a feedforward neural network for feature extraction. The output of the feedforward neural network is added to the encoded output of the self-attention to obtain the final encoding output of the model, as shown in Equation (16).

$$X = A + ((ReLU(AW_1 + b_1))W_2 + b_2) \tag{16}$$

ISSN 2616-5775 Vol. 6, Issue 6: 8-15, DOI: 10.25236/AJCIS.2023.060602

In the equation, $W_1, W_2, b_1,$ and b_2 are all learnable parameters of the fully connected layer.

2.3. Decoding Layer

The decoding layer employs conditional random fields to perform decoding. Assuming that the output sequence of the encoding layer is $\mathrm{X} = (\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_l)$ and the predicted label sequence is $Y = (y_1, y_2, \ldots, y_l)$, the score function of the predicted label sequence is given by Equation (17).

$$Score(X,Y) = \sum_{i=1}^{l}(W_{y_{i-1},y_i} + P_{i,y_i}) \tag{17}$$

Where l represents the number of predicted characters, and $\mathrm{W}_{y_{\mathrm{i}-1},y_{\mathrm{i}}}$ refers to the transfer probability from the predicted label $y_{\mathrm{i}-1}$ to the predicted label y_{i}, which will be updated with the learning process. $\mathrm{P}_{\mathrm{i},y_{\mathrm{i}}}$ stands for the probability of the label y_i corresponding to the ith character of the text sequence. The probability partially derived from the output of the encoding layer constitutes the state function.

Given a known input sequence X, the calculation formula for predicting the label sequence Y is shown in Equation (18).

$$P(Y|X) = \frac{exp(Score(X,Y))}{\sum_{Y' \in Y_x} exp(Score(X,Y'))} \tag{18}$$

Where Y_x represents the true label sequence, and the log-likelihood function of the predicted label sequence is obtained by taking the logarithm of both sides of the equation as shown in Equation (19). The global optimal solution is realized during the model learning process by the log-likelihood function maximization method.

$$logP(Y|X) = Score(X,Y) - log\left(\sum_{Y' \in Y_x} exp\left(Score(X,Y')\right)\right) \tag{19}$$

3. Experiment

In order to evaluate the performance and effectiveness of the proposed Chinese named entity recognition model, experiments were conducted on the CCL2021 dataset.

3.1. Experiment Settings

The present experiment was implemented by defining the key hyperparameters as presented in Table 1. Notably, distinct learning rate parameters were assigned to the feature extraction module and the backbone network. To optimize the learning parameters and boost the model's performance, the stochastic gradient descent (SGD) algorithm was adopted. Specifically, for the partial radical feature extraction module and the pinyin feature extraction module, a one-dimensional convolutional neural network with 256 convolutional kernels was utilized for feature extraction, with the size of each convolutional kernel set as 1, and the stride as 3.

Table 1: Hyperparameters Settings

Hyperparameters	Values
batch_size	16
epochs	100
char_embeddings_dim	50
word_embeddings_dim	50
radical_embedding_dim	128
pinyin_embedding_dim	128
learning_rate	4e-5
radical_ learning_rate	6e-4
pinyin_ learning_rate	5e-4
attention_heads_num	8
attention_heads_dim	32

ISSN 2616-5775 Vol. 6, Issue 6: 8-15, DOI: 10.25236/AJCIS.2023.060602

3.2. Performances

The specific results of experiments to verify the effectiveness of named entity recognition using various models were summarized in Table 2.

Table 2: Main Results on CCL2021

Models	P	R	F1
Lattice-LSTM[9]	87.42%	88.36%	87.89%
Softlexicon(LSTM)[1]	88.54%	88.29%	88.29%
LR-CNN[15]	88.25%	88.57%	88.41%
FLAT[10]	90.30%	91.02%	90.66%
Our Model	**91.01%**	**92.28%**	**91.75%**

According to Table 2, the FLAT model that integrates both pinyin and partial radical features achieved an F1 score 3.86% higher than the Lattice-LSTM model. It also outperformed the LR-CNN and Softlexicon (LSTM) models, which only consider improved word-level information, by 3.34% and 3.46%, respectively. Additionally, the FLAT model exhibited improvement in the F1 score from 90.66% to 91.75% compared to its single-feature counterpart. It was demonstrated that the integration of multiple feature information in the FLAT model can effectively enhance performance of named entity recognition for Chinese medical texts.

4. Conclusions

In this paper, a FLAT model integrating pinyin and partial radical features is proposed to address the issue of insufficient semantic information extraction within Chinese characters. By introducing both partial radical and pinyin features in the embedding layer, the proposed model enhances the semantic information of input sentences and increases the accuracy of named entity recognition. Furthermore, the effectiveness of the model is certified by comparing its performance with state-of-the-art named entity recognition algorithms based on word-level information in the medical text domain.

References

[1] Ma, R., Peng, M., Zhang, Q., Wei, Z. and Huang, X., Simplify the Usage of Lexicon in Chinese NER; proceedings of the Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, F July, 2020. Association for Computational Linguistics.

[2] Yang, J., Teng, Z., Zhang, M. and Zhang, Y., Combining discrete and neural features for sequence labeling; proceedings of the Computational Linguistics and Intelligent Text Processing: 17th International Conference, F, 2018.

[3] He, H. and Sun, X., A unified model for cross-domain and semi-supervised named entity recognition in Chinese social media; proceedings of the Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, F, 2017.

[4] Tian, Y., Song, Y., Xia, F., Zhang, T. and Wang, Y., Improving Chinese word segmentation with wordhood memory networks; proceedings of the Proceedings of the 58th annual meeting of the association for computational linguistics, F, 2020.

[5] Lu, Y., Zhang, Y. and Ji, D., Multi-prototype Chinese character embedding; proceedings of the Proceedings of the tenth international conference on language resources and evaluation (LREC'16), F, 2016.

[6] Dong, C., Zhang, J., Zong, C., Hattori, M. and Di, H., Character-based LSTM-CRF with radical-level features for Chinese named entity recognition; proceedings of the The 5th Conference on Natural Language Processing and Chinese Computing & The 24th International Conference on Computer Processing of Oriental Languages, F, 2016.

[7] Liu, Z., Zhu, C. and Zhao, T., Chinese named entity recognition with a sequence labeling approach: based on characters, or based on words?; proceedings of the Proceedings of the Advanced intelligent computing theories and applications, and 6th international conference on Intelligent computing, F, 2010.

[8] Li, H., Hagiwara, M., Li, Q. and Ji, H., Comparison of the Impact of Word Segmentation on Name Tagging for Chinese and Japanese; proceedings of the Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), F May, 2014.

[9] Zhang, Y. and Yang, J., Chinese NER Using Lattice LSTM; proceedings of the Proceedings of the

ISSN 2616-5775 Vol. 6, Issue 6: 8-15, DOI: 10.25236/AJCIS.2023.060602

56th Annual Meeting of the Association for Computational Linguistics, F July, 2018.
[10] Li, X., Yan, H., Qiu, X. and Huang, X., FLAT: Chinese NER Using Flat-Lattice Transformer; proceedings of the Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, F July, 2020. Association for Computational Linguistics.
[11] Zhu, W., Jin, X., Ni, J., Wei, B. and Lu, Z. (2018) Improve word embedding using both writing and pronunciation. PloS one, 13 (12): e0208785.
[12] Chen, A. and Yin, C., CRW-NER: Exploiting Multiple Embeddings for Chinese Named Entity Recognition; proceedings of the 2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD), F, 2021. IEEE.
[13] Yang, J., Wang, H., Tang, Y. and Yang, F., Incorporating lexicon and character glyph and morphological features into BiLSTM-CRF for Chinese medical NER; proceedings of the 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE), F, 2021. IEEE.
[14] Zhang, Y., Liu, Y., Zhu, J., Zheng, Z., Liu, X., Wang, W., Chen, Z. and Zhai, S., Learning Chinese word embeddings from stroke, structure and pinyin of characters; proceedings of the Proceedings of the 28th ACM International Conference on Information and Knowledge Management, F, 2019.
[15] Gui, T., Ma, R., Zhang, Q., Zhao, L., Jiang, Y.-G. and Huang, X., CNN-Based Chinese NER with Lexicon Rethinking; proceedings of the ijcai, F, 2019.