English Text Named Entity Recognition Method by Fusing Local and Global Features

Liuxin Gao[1]

School of Foreign Languages, Zhengzhou University of Science and Technology
450064 Zhengzhou, China
publicgj@163.com

Received September. 5, 2024; Revised and Accepted September. 17, 2024

Abstract. Because of the ambiguity and dynamic nature of natural language, the research of named entity recognition is very challenging. As an international language, English plays an important role in the fields of science and technology, finance and business. Therefore, the early named entity recognition technology is mainly based on English, which is often used to identify the names of people, places and organizations in the text. International conferences in the field of natural language processing, such as CoNLL, MUC, and ACE, have identified named entity recognition as a specific evaluation task, and the relevant research uses evaluation corpus from English-language media organizations such as the Wall Street Journal, the New York Times, and Wikipedia. The research of named entity recognition on relevant data has achieved good results. Aiming at the sparse distribution of entities in text, a model combining local and global features is proposed. The model takes a single English character as input, and uses the local feature layer composed of local attention and convolution to process the text pieceby way of sliding window to construct the corresponding local features. In addition, the self-attention mechanism is used to generate the global features of the text to improve the recognition effect of the model on long sentences. Experiments on three data sets, Resume, MSRA and Weibo, show that the proposed method can effectively improve the model's recognition of English named entities.

Keywords: English named entity recognition, Local feature, Global feature, Self-attention mechanism, Long sentence.

1. Introduction

The concept of Named Entity originated in the field of natural language processing to define proper nouns that appear in text and have a specific meaning. Usually, the narrow named entity includes the specific person, place name and organization name. With the deepening of related research, the scope of named entities is also expanding. For example, in the research of named entities in the fields related to molecular biology, subject-related named entities such as protein names, catalyst names and molecular structure descriptors have attracted more attention from researchers [1-3]. In the medical field, named entities such as disease names, conditions, and body parts are of even greater research [4-7].

The named entities related to these fields have greatly broadened the research scope of named entity recognition, enriched the application scenarios of named entity recognition, and made the research in this field have extensive and in-depth development. The research on named entity recognition started earlier. After years of development, many scholars have produced fruitful research results from different perspectives and different methods. According to the main techniques used in the research, there are three main methods for named entity recognition [8,9].

1. Based on rule matching. That is, entity identification is carried out using the matching rules made by human. Such as extracting entities from the original text. Tapaswi et al. [10] proposed a matching rule based on part-of-speech tagging information. In the field of biomedicine, Chauhan et al. [11] used pre-processed medical dictionaries combined with corresponding rules to identify entities such as proteins and genes. Manual construction of matching rules requires a lot of time and effort, and can only be applied to specific entity types and languages, and the limitations are very obvious. After matching rules are specified for tags such as keywords, indicators, and punctuation marks, the matching mode is based on the string.
2. Based on statistical machine learning. That is, a supervised learning algorithm is used to train a model on a large amount of labeled data. The text content will be transformed into the corresponding feature representation, and the corresponding entity annotation can be obtained after input to the model. Chen et al. [12] used the maximum entropy model to build a named entity recognition model suitable for multiple languages, and achieved good results in the English and German named entity recognition evaluation tasks of CoNLL-2003. Conditional random field can consider the context information in calculation, which becomes a big tool to

solve the task of named entity recognition. It has been widely used in named entity recognition tasks in different fields such as biomedicine and chemistry [13]. Sharma et al. [14] used two-layer conditional random fields and took the output of the first layer as the input of the second layer. Experimental results showed that the second layer conditional random field could use the implicit representation learned by the first layer conditional random field, and the performance was better than that of the single layer model. Although statistical machine learning methods have shown good results and interpretability in many scenarios, the construction of appropriate text features is a complex task that requires professional knowledge, domain knowledge and a large amount of human input, thus restricting the further development and application of related research.

3. Based on deep learning. Deep learning methods can automatically learn deep implicit information, eliminating the tedious step of manually designing matching rules or text features. The greatly improved hardware performance in recent years has also alleviated the problem of long training times for deep learning models to a certain extent. More and more scholars use deep learning methods to solve named entity recognition problems.

Recurrent Neural Network (RNN) is a common neural network structure that is suitable for modeling sequential problems. Models based on RNNS and their related variants, such as Gated Recurrent units (GRU) [15], Long Short-Term Memory networks (LSTM) [16], are widely used in this field. Yin et al. [17] proposed the Bi-LSTM model, which used two-way LSTM to learn the feature representation of the two directions before and after sentences, and achieved the best recognition effect at that time, becoming a classical model structure for solving sequence annotation problems. VeeraSekharReddy et al. [18] applied the Bi-LSTM model to letter features, so that the model could obtain excellent results without adding any artificial construction features, and could be transferred to multiple languages. In theory, RNN-based models can handle sequences of arbitrary length and capture long distance dependencies in sentences well, but in the course of practical model training, too long input sequences often lead to gradient disappearance. The dependency of RNN structure makes it difficult to parallelize the training and derivation process of the model, and the computational efficiency is relatively low. Convolutional Neural networks (CNNs) are another widely used Neural Network structure that creatively introduces convolutional operations into artificial neural networks, often used to extract local features in sentences. Yu et al. [19] used convolutional layer to extract contextual features of each word and combined with attention mechanism to carry out entity extraction. Liu et al. [20] proposed a GRN model based on CNN, which fused local features of each word into global features through a gated relational network. Experimental results showed that GRN was better than RNN in capturing long-distance dependencies.

Based on the above analysis, a named entity recognition model is proposed, which uses local attention and convolution operations to extract text local features, and then extracts global features with self-attention. The model takes a single English character as input, and can differentiate the input content according to the characteristics of the named entity recognition task. Combined with the fully connected property of multi-head attention in the global feature layer, the model can effectively improve the effect of English entity recognition.

2. Problem Description

Due to its unique chain loop structure, recurrent neural networks (RNN) are suitable for the modeling of sequential data such as speech and text [21,22]. In particular, Xiao et al. [23] proposed a model based on BiLSTM, which was widely used in named entity recognition tasks and had achieved good results. However, the current named entity recognition model based on RNN still has the following shortcomings:

1. Existing studies have shown that models based on RNNs will produce the phenomenon of ”forgetting”, that is, the model will lose the information learned earlier [24]. The main reason is that RNN needs to repeat the same calculation process on a very long time series, and the calculation graph will become extremely deep, resulting in the problem of gradient disappearance and gradient explosion, which makes the optimization of the model extremely difficult. Therefore, as the length of sentences increases, the RNN will gradually forget the previously stored historical information, causing the model to process long sentences less effectively. In some cases, named entity recognition tasks need to capture long-distance dependencies in sentences to help the model classify entities correctly.
2. Under normal circumstances, most of the content of a sentence is composed of non-entity content such as verbs, conjunctions, function words, etc., and the named entities that really need to be identified usually only account for a small part of the whole sentence, that is, the distribution of entities in the text is sparse. However, the common named entity recognition model treats all inputs indiscriminately in the process of modeling, thus introducing a lot of redundant information, which will interfere with the prediction results. Therefore, in the actual modeling process, we should pay more attention to the entity-related part of the sentence, while

ignoring the interference of other secondary information, so as to improve the model's recognition effect of named entities.

Based on the above analysis, on the basis of Bi-LSTM model architecture, an improved feature extraction method is proposed, which uses local Attention and Convolution operations to extract local features, and then Self-attention to extract global features Identify the model. There are two major improvements in the new model:

1. In the global coding phase, the global feature layer based on multi-head self-attention mechanism is used to replace the chained Bi-LSTM layer. Using the fully connected characteristics of multi-head self-attention, the problem of ”forgetting” caused by cyclic repetition is alleviated, and the processing ability of the model on long sentences is improved.
2. The local feature layer is added to the model, which is mainly used to extract the local features of text using local attention and one-dimensional convolution inside several equal-length Windows. This layer can dynamically assign different weights to each input according to the characteristics of the named entity recognition task, highlight the entity-related content, and further synthesize the context content to generate local feature coding to improve the predictive performance of the model.

3. Feature Construction Method

The new model uses local and global features to identify Chinese named entities. The main difference between the two is that the length of text is different in the construction of features. Local features mainly construct features on text fragments, while global features consider the entire input sequence. This section describes the construction methods of two features.

3.1. Local Feature Construction Method

When you see some pictures, the visual focus is often attracted by the speed limit sign in front of you, and the trees and roads that occupy the main part of the picture are easy to be ignored. Named entity recognition tasks are similar in that, whether dealing with images or text, what is really valuable is usually only a small part of the whole. Entities are sparsely distributed in the text, i.e. a sentence is usually composed of a large number of non-entity words and a small number of entity words [25-28]. Therefore, the named entity recognition model does not need to treat all input characters equally in the process of sentence processing, but should highlight a certain part of the sentence content. Text content can also be processed in the same way as image content, focusing only on the important parts and ignoring the minor content. Therefore, a local feature layer is added to the model in this paper, and local attention and convolution operations are used to construct local features on text fragments.

The first step of local feature construction is to calculate the corresponding weight value of each input through the local attention mechanism, and adjust each input dynamically according to these weight values. If the corresponding weight value of an input is large, it means that its corresponding content needs to be emphasized. Otherwise, its contents should be appropriately ignored. Figure 1 shows the detailed calculation of partial attention.

As shown in Figure 1, the scope of the local attention mechanism is a window, each window corresponds to a text fragment. To reduce computational complexity, the length of the window takes a fixed odd value of $l = 2m{+}1, m \in N$, and receives a vectorized text representation as input. If the vector corresponding to c in the middle of the window is denoted as x, then all vectors in the window can be represented as $T = x_{c-m}, \cdots, x_c, \cdots, x_{c+m}$. For any vector x_k in T, the attention fraction a_k corresponding to x_k and any context vector x_c can be calculated according to equation (1), where W_1, W_2, v are trainable model parameters.

$$a_k = v^T \tanh(W_1 x_c + W_2 x_k), k \in [c-m, c+m]. \tag{1}$$

$$p_k = softmax(a_k), k \in [c-m, c+m]. \tag{2}$$

$$h_k = p_k \circ x_k, k \in [c-m, c+m]. \tag{3}$$

After obtaining a set of attention scores $a_k | k \in [c-m, c+m]$, enter them into the softmax function shown in equation (2) and calculate the attention weight p_x for each vector in the window.

Finally, this group of weight values p_k is multiplied by the corresponding vector in window T by equation (3) to obtain the adjusted window content, denoted as $H = h_{c-m}, \cdots, h_c, \cdots, h_{c+m}$. And so on, the window moves

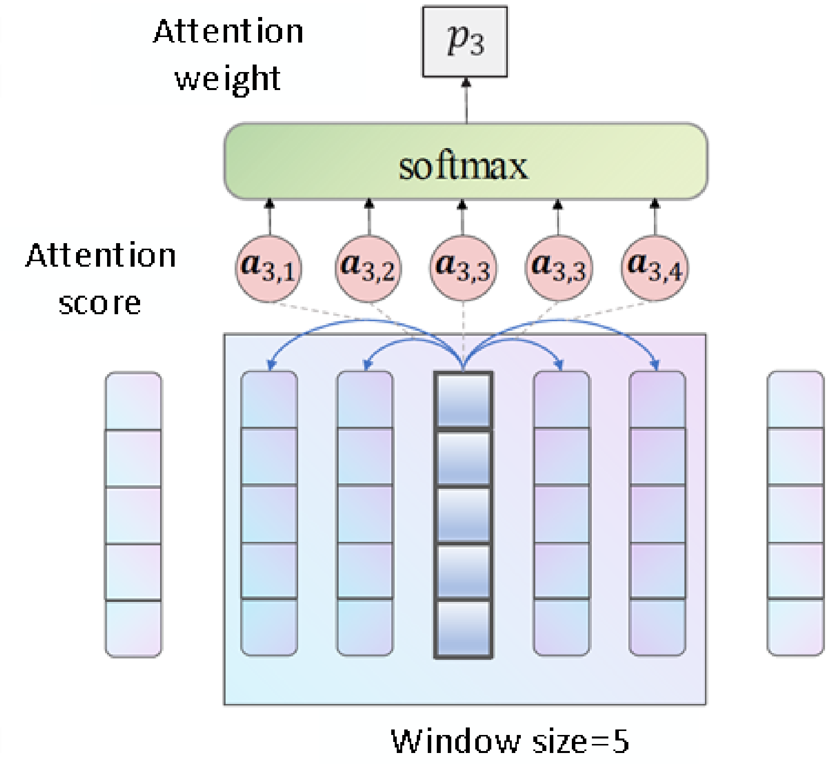

Fig. 1. Local attention calculation process

back one character at a time from the starting position of the sentence and performs the same computation at the new position until the entire sentence has been processed.

The second step in constructing local features is a text convolution operation. After the attention weight corresponding to each input vector in the window is dynamically adjusted using the sliding window, the output result H is then convolution operation. The calculation process is shown in Figure 2. In each convolution operation, the convolution kernel mainly acts on the window content H adjusted by the local attention mechanism. The convolution kernel width is the same as the size of the above window, still l. In order to reduce the model parameters and speed up the operation [29,30], the convolution results are compressed by average pooling. After the compressed results are input into the activation function, the corresponding local feature representation is obtained. Formula (4) is a method for computing the local eigenrepresentation O_c, where R is the convolution kernel and the "$\odot$" symbol represents the Hadamard product of the matrix. f is the ReLU activation function. The convolution kernel only moves in one direction, and each time it moves, the convolution calculation process is repeated until the entire sentence is computed.

$$O_c = f(\frac{1}{l}\sum_{l}[R \odot H]). \tag{4}$$

The dynamic method is used to calculate the weight of local attention, and the parameters related to the operation are adjusted in the process of model training, so that the weight allocation of embedded vector is more in line with the needs of named entity recognition task. The convolution operation takes all the contents in the window into consideration and extracts the local feature representation from the text fragment by means of convolution. In this way, the calculated results fully take into account the relationship between one successive input in the window and can effectively capture the important local features in the text. The resulting local feature representation is denoted as $O = o_1, o_2, \cdots, o_n$.

3.2. Global Feature Construction Method

At present, there are many named entity recognition models based on RNNS to encode the whole sequence, especially bidirectional RNNS can encode the sequence from the front to the back and from the back to the front. It has made remarkable achievements in sequential data processing, and is the first choice model to solve many

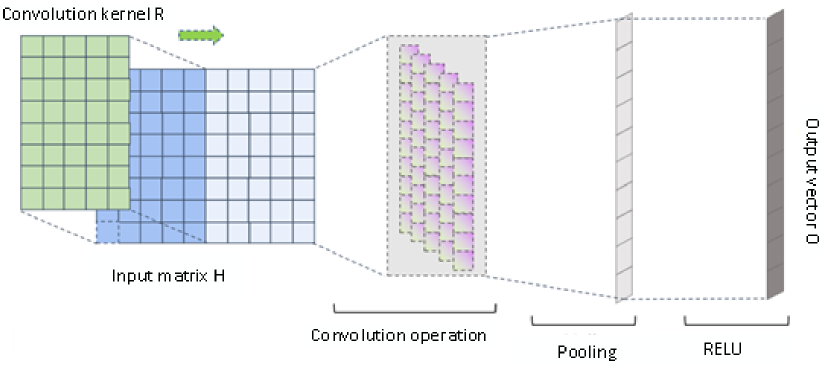

Fig. 2. Convolution computation procedure

language problems. However, the model based on RNN has a serious problem of disappearing gradient, which leads to a significant decline in entity recognition on long sentences. Considering the fully connected nature of the multi-head self-attention mechanism [31], each input takes into account all other inputs when calculating attention, this pin-to-pair connected property can capture dependencies of any length. At the same time, setting multiple attention heads can enable the model to extract features in different linear Spaces at the same location, and improve the representation ability of the model through different levels of information. Therefore, the multi-head self-attention mechanism is used to model the whole sequence in order to extract the global coding of the sequence.

As shown in Formula (5-7), the global feature encode receives the local feature $O = o_1, o_2, \cdots, o_n$ constructed in the previous section as input, and each attentional head m will pass through three different mapping matrices W_Q^m, W_K^m and W_V^m will linearly transform O to obtain the corresponding query matrix Q^m, key matrix K^m and value matrix V^m.

$$Q^m = W_Q^m O. \tag{5}$$

$$K^m = W_K^m O. \tag{6}$$

$$V^m = W_V^m O. \tag{7}$$

Calculate the self-attention t_m of the current head according to equation (8). The value of the scaling factor d is equal to the dimension of each attention head.

$$t_m(Q^m, K^m, V^m) = softmax(\frac{Q^m (K^m)^T}{\sqrt{d}})V^m. \tag{8}$$

Each self-attentional head can extract different features from different linear Spaces. After the above calculation process is repeated, the self-attention $t_1, \cdots, t_h$ of h head is obtained successively, and then the results are spliced and input a fully connected layer W^z to obtain the final global feature representation $Z = z_1, z_2, \cdots, z_n$.

$$Z = W^z(t_1 \oplus t_2 \oplus \cdots \oplus t_n). \tag{9}$$

3.3. Proposed Model Structure

This section mainly introduces the concrete structure of the proposed model. Local features and global features are integrated in series, effectively preserving the advantages of the two features. The model in this paper mainly consists of the following four sub-parts:

1. The embedding layer. In Western languages typical of English and French, word is the basic unit of semantic expression, so many related work of named entity recognition is carried out on the basis of word. However, since the English text does not have explicit word separators, if the processing mode of other languages is directly applied, the text content must be divided by relevant tools first, and the result of word segmentation will have a direct impact on the subsequent processing process. Therefore, English named entity recognition tasks

are usually carried out at word granularity to avoid error propagation caused by wrong word segmentation results.

In order to input English characters into the model, it is necessary to convert the characters into corresponding vector forms, and the vectorized character representation is easy to train and derive the deep learning model. The vector representation obtained by pre-training from a large number of corpus contains richer semantic information than the vector representation generated by random initialization, which can accelerate the convergence speed of the model and improve the prediction effect of the model [50]. The specific embedding process of the embedded layer of the model in this paper is as follows:

$$x_i = Emb(c_i). \tag{10}$$

Let $S = c_1, c_2, \cdots, c_n$, represent an input sequence of length n, where $c_i \in S$ represents the $i - th$ character in the sequence. For each character c_i, the formula (10) is used to convert it into the corresponding vector representing x_i, where Emb represents the pre-trained vector lookup table. The n characters $c_1, \cdots, c_n$ in the sequence S are mapped to n corresponding vectors $x_1, \cdots, x_n$, respectively, to form a two-dimensional matrix $X = x_1, \cdots, x_n$, matrix X is the vectorized representation of the sequence S.

2. Local feature layer. This part mainly consists of two parts: local attention layer and convolution layer. The local attention layer receives the output from the embed layer and assigns the input to a small window for processing. According to the relative position of the vector in the window, the intermediate vector and the context vector can be determined, and the attention weight corresponding to the intermediate vector can be calculated from the context vector. By dynamically adjusting the size of attention weight value, the large amount of noise caused by non-physical content is reduced. Subsequent convolution layers perform convolution operations using convolution check inputs of the same size as the above window to extract local features $O = o_1, o_2, \cdots, o_n$ from text fragments, thus extending the single word embedding representation to include richer contextual information.
3. Global feature layer. The traditional named entity recognition model based on RNNS is not effective when dealing with long sentences. Inspired by the work of Vaswani et al., the global feature layer encodes the entire input sequence using a multi-head self-attention mechanism to obtain the global feature corresponding to the entire sentence $Z = z_1, \cdots, z_n$.
4. The decoding layer. The final layer of the model is calculated using conditional random fields based on the output of the global feature layer, and the final annotation result is obtained. Compared with other decoding methods, conditional random field can make full use of the global context features to model the dependency between labels, thus improving the accuracy of the annotation results. Therefore, the labeled sequence is obtained by using linear chain random fields.

4. Experiment and Analysis

The experiment used two publicly available named entity recognition datasets: Weibo and MSRA. Statistics related to the data set are listed in Table 1.

Table 1. Data set statistics

Data set	Type	Training set	Validation set	Test set
Weibo	Number of sentences	1400	270	270
Weibo	Number of characters	73800	14500	14800
Weibo	Entity number	1890	420	390
MSRA	Number of sentences	45000	2600	3400
MSRA	Number of characters	2171500	8700	172600
MSRA	Entity number	75100	5800	6200

In addition to the Bi-LSTM model based on word embeddings, the comparison model selected in this section also cites some other research results in the field of English named entity recognition in recent years for comparison, and the data listed are all from the original literature. The comparison objects mainly include Transformer [32], STHN model [33], CNN-BIRNN model[34], SoftLexicon model [35], and AT4CNER model [36].

The test results on the MSRA dataset are shown in Table 2. Compared with the baseline model, the proposed model achieves an improvement of 2.53%, 4.33% and 3.46% in three evaluation indexes, respectively.

The test results on the Weibo dataset are shown in Table 3. Compared with the baseline model, the model in this paper has achieved an improvement of 4.32%, 0.94% and 2.33% respectively in the three evaluation indexes

Table 2. Test results on the MSRA dataset

Model	Precision/%	Recall/%	F1/%
Bi-LSTM	90.85	87.07	88.92
Transformer-P	91.23	90.64	90.46
SoftLexicon	91.87	91.33	90.59
CNN-BiRNN	92.15	91.32	91.78
AT4CNER	91.84	89.69	90.75
Proposed	93.38	91.40	92.38

of P, R and F1 value. It can be seen from the experimental data that the evaluation results of the Weibo dataset are significantly lower than those of the other two datasets, indicating that named entity recognition in the microblog book is still a very difficult task. This is mainly due to the fact that the original corpus of the Weibo dataset comes from Sina Weibo, which contains a large number of emojis and hyperlinks, and the grammar and wording of the corpus itself are not as standardized as the other two data sets from resume texts and news reports. These factors increase the difficulty of feature extraction and affect the final recognition effect.

Table 3. Test results on the Weibo dataset

Model	Precision/%	Recall/%	F1/%
Bi-LSTM	60.97	51.56	55.87
Transformer-P	61.87	60.34	60.63
SoftLexicon	61.58	60.88	61.15
CNN-BiRNN	60.11	53.73	56.74
AT4CNER	55.83	50.79	53.19
Proposed	65.29	52.50	58.20

5. Conclusion

In this paper, two shortcomings of the existing named entity recognition model based on RNN are analyzed, and an improved model is proposed. The new model uses fully connected multi-head self-attention instead of RNN to model sentence sequences, which improves the model's performance on long sentences. At the same time, it uses the local feature layer to dynamically adjust the weight of the input vector in the window and fully integrate the context content, which further improves the recognition effect of the named entity recognition model. Experimental results on the MSRA and Weibo named entity recognition data sets show that the F1 value of the new model is 3.46% and 2.33% higher than that of the Bi-LSTM model, respectively, and the performance is more stable when dealing with inputs of different lengths.

6. Conflict of Interest

The authors declare that there are no conflict of interests, we do not have any possible conflicts of interest.

Acknowledgments. This study was phased research result of the Reform and Practice of Practical Teaching System for Applied Translation Undergraduate Majors from the Perspective of Technology Hard Trend of Henan Province Education Reform Project in 2024 (Project number: 2024SJGLX0581) and The Key Teaching Reform Project of Zhengzhou University of Science and Technology in 2024, ”Innovative Research on Practical Teaching of Digital-Intelligence Technology Enabling Production-Teaching Integration” (Project number: 2024JGZD11).

References

1. Medileh S, Laouid A, Hammoudeh M, et al. A Multi-Key with Partially Homomorphic Encryption Scheme for Low-End Devices Ensuring Data Integrity[J]. Information, 2023, 14(5): 263.
2. Yan X, Zhou G, Huang Y, et al. Secure estimation using partially homomorphic encryption for unmanned aerial systems in the presence of eavesdroppers[J]. IEEE Transactions on Intelligent Vehicles, 2024.

3. Yu J, Lu Z, Yin S, et al. News recommendation model based on encoder graph neural network and bat optimization in online social multimedia art education[J]. Computer Science and Information Systems, 2024 (00): 25-25.
4. S. Yin, H. Li, Y. Sun, M. Ibrar, and L. Teng. Data Visualization Analysis Based on Explainable Artificial Intelligence: A Survey[J]. IJLAI Transactions on Science and Engineering, vol. 2, no. 2, pp. 13-20, 2024.
5. Bharadiya J. A comprehensive survey of deep learning techniques natural language processing[J]. European Journal of Technology, 2023, 7(1): 58-66.
6. Chiruzzo L, Jimnez-Zafra S M, Rangel F. Overview of IberLEF 2024: Natural Language Processing Challenges for Spanish and other Iberian Languages[C]//Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2024), co-located with the 40th Conference of the Spanish Society for Natural Language Processing (SEPLN 2024), CEUR-WS. org. 2024.
7. Yin S, Li H, Laghari A A, et al. An anomaly detection model based on deep auto-encoder and capsule graph convolution via sparrow search algorithm in 6G internet-of-everything[J]. IEEE Internet of Things Journal, vol. 11, no. 18, pp. 29402-29411, 2024. DOI: 10.1109/JIOT.2024.3353337.
8. Esfahani M N. Content Analysis of Textbooks via Natural Language Processing[J]. American Journal of Education and Practice, 2024, 8(4): 36-54.
9. Treviso M, Lee J U, Ji T, et al. Efficient methods for natural language processing: A survey[J]. Transactions of the Association for Computational Linguistics, 2023, 11: 826-860.
10. Tapaswi N. An efficient part-of-speech tagger rule-based approach of Sanskrit language analysis[J]. International Journal of Information Technology, 2024, 16(2): 901-908.
11. Chauhan S, Shet J P, Beram S M, et al. Rule based fuzzy computing approach on self-supervised sentiment polarity classification with word sense disambiguation in machine translation for Hindi language[J]. ACM Transactions on Asian and Low-Resource Language Information Processing, 2023, 22(5): 1-21.
12. Chen Y, Wu L, Zheng Q, et al. A boundary regression model for nested named entity recognition[J]. Cognitive Computation, 2023, 15(2): 534-551.
13. XIE X, XIE Z, MA K, et al. Geological named entity recognition combined BERT and BiGRU-Attention-CRF model[J]. Geological Bulletin of China, 2023, 42(5): 846-855.
14. Sharma S, Prabha S, Singh V. Constructing Conditional Random Fields for Automated Land Cover Mapping from Hyper Spectral Images[C]//2024 International Conference on Optimization Computing and Wireless Communication (ICOCWC). IEEE, 2024: 1-6.
15. Jiang Y, Yin S. Heterogenous-view occluded expression data recognition based on cycle-consistent adversarial network and K-SVD dictionary learning under intelligent cooperative robot environment[J]. Computer Science and Information Systems, 2023, 20(4): 1869-1883.
16. Woodbridge J, Anderson H S, Ahuja A, et al. Predicting domain generation algorithms with long short-term memory networks[J]. arxiv preprint arxiv:1611.00791, 2016.
17. Yin X, Liu Z, Liu D, et al. A Novel CNN-based Bi-LSTM parallel model with attention mechanism for human activity recognition with noisy data[J]. Scientific Reports, 2022, 12(1): 7878.
18. VeeraSekharReddy B, Rao K S, Koppula N. An attention based bi-LSTM DenseNet model for named entity recognition in english texts[J]. Wireless Personal Communications, 2023, 130(2): 1435-1448.
19. VeeraSekharReddy B, Rao K S, Koppula N. An attention based bi-LSTM DenseNet model for named entity recognition in english texts[J]. Wireless Personal Communications, 2023, 130(2): 1435-1448.
20. Liu L, Wang Y, Peng J, et al. GLR-CNN: CNN-based Framework with Global Latent Relationship Embedding for High-resolution Remote Sensing Image Scene Classification[J]. IEEE Transactions on Geoscience and Remote Sensing, 2024.
21. Yin S, Wang L, Shafiq M, et al. G2Grad-CAMRL: an object detection and interpretation model based on gradient-weighted class activation mapping and reinforcement learning in remote sensing images[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2023, 16: 3583-3598.
22. Wang L, Shoulin Y, Alyami H, et al. A novel deep learning-based single shot multibox detector model for object detection in optical remote sensing images [J]. Geoscience Data Journal, vol. 11, no. 3, pp. 237-251, 2024.
23. Xiao M, Yang B, Wang S, et al. GRA-Net: Global receptive attention network for surface defect detection[J]. Knowledge-Based Systems, 2023, 280: 111066.
24. Orvieto A, Smith S L, Gu A, et al. Resurrecting recurrent neural networks for long sequences[C]//International Conference on Machine Learning. PMLR, 2023: 26670-26698.
25. Liu S, Zhao Y, An Y, et al. GLFANet: A global to local feature aggregation network for EEG emotion recognition[J]. Biomedical Signal Processing and Control, 2023, 85: 104799.
26. Zhao X, Cheah C C. BIM-based indoor mobile robot initialization for construction automation using object detection[J]. Automation in Construction, 2023, 146: 104647.
27. Zhu P, Hou X, Tang K, et al. Unsupervised feature selection through combining graph learning and l-norm constraint[J]. Information Sciences, 2023, 622: 68-82.
28. Zhao Y, Li H, Yin S. A multi-channel character relationship classification model based on attention mechanism[J]. Int. J. Math. Sci. Comput.(IJMSC), 2022, 8: 28-36.
29. Soni S, Chouhan S S, Rathore S S. TextConvoNet: A convolutional neural network based architecture for text classification[J]. Applied Intelligence, 2023, 53(11): 14249-14268.
30. Umer M, Imtiaz Z, Ahmad M, et al. Impact of convolutional neural network and FastText embedding on text classification[J]. Multimedia Tools and Applications, 2023, 82(4): 5569-5585.
31. Wu Y, Kong Q, Lai Y, et al. CDText: scene text detector based on context-aware deformable transformer[J]. Pattern Recognition Letters, 2023, 172: 8-14.

32. Han D, Pan X, Han Y, et al. Flatten transformer: Vision transformer using focused linear attention[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2023: 5961-5971.
33. Vasu P K A, Gabriel J, Zhu J, et al. FastViT: A fast hybrid vision transformer using structural reparameterization[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 5785-5795.
34. Zhou H Y, Guo J, Zhang Y, et al. nnformer: Volumetric medical image segmentation via a 3d transformer[J]. IEEE Transactions on Image Processing, 2023.
35. Gao S, Zhou C, Zhang J. Generalized relation modeling for transformer tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 18686-18695.
36. Lin W, Wu Z, Chen J, et al. Scale-aware modulation meet transformer[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 6015-6026.

Biography

Liuxin Gao is with the School of Foreign Languages, Zhengzhou University of Science and Technology. Several papers had been published related to the work in this paper. Research direction is English text analysis, English data analysis.