IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 14 October 2023, accepted 7 November 2023, date of publication 14 November 2023,
date of current version 20 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3332994

RESEARCH ARTICLE

DTDM: Dynamic Temporal Convolutional Network and Dynamic Multihead Attention for Chinese Named Entity Recognition

YUAN HUANG, YANXIA LI, AND XIAOYU ZHANG
School of Information and Electrical Engineering, Hebei University of Engineering, Handan 056038, China
Corresponding author: Yanxia Li (liyxhebeu2021@163.com)

ABSTRACT Compared with English Named Entity Recognition (NER), Chinese Named Entity Recognition (CNER) has a high difficulty in word segmentation, and accurate extraction of contextual semantic feature information is a key work of CNER. For that, we propose a CNER model to extract both local and global contextual semantic feature information. First, we propose to apply the dynamic convolutional kernel to the convolutional layer of TCN to enhance the local features of contextual semantic feature information. Second, we define a dynamic scaling factor computation method to compute the correlation between named entity characters in the multihead attention, which to process the problem of sparse distribution of named entities, and can efficiently extract the global features of contextual semantics. We validated the effectiveness of the proposed model on the Weibo dataset with an F1 value of 89.24%, which is better than commonly used models.

INDEX TERMS Chinese named entity recognition, multihead attention, temporal convolutional network, dynamic convolutional network, conditional random field.

I. INTRODUCTION

Named Entity Recognition (NER) is the basic task of Nature Language Processing (NLP) and the upstream task of various tasks in the field of NLP. The result of NER task has a significant impact on various downstream tasks such as relationship extraction, entity link, information extraction, intelligent question answering, etc. [1]. The goal of the NER task is to find named entities from the text and divide them into different types, such as people's names, place names, and institutional names. The Chinese Named Entity Recognition(CNER) is regarded as the text sequence label prediction.

Early named entity recognition methods experienced two stages: rule-based and manual features and statistical machine learning. The basic idea of rule-based and manual feature methods is matching rules, which mainly relies on dictionaries, manual templates,and regular expressions, and has poor transfer ability. The method based on statistical machine learning mainly uses a large corpus to train model. And extensive manual feature engineering is performed for different tasks to design appropriate feature templates. Although the method solves the transfer ability problem based on rules and the manual feature method, building a large number of manual feature templates is still time-consuming.

With the development of deep learning neural networks, deep learning methods do not require manually setting rules and extracting features, and they outperform traditional named entity recognition methods. The Long Short Term Memory (LSTM) and Conditional Random Field (CRF) (LSTM-CRF) [2] is used as the baseline model in the NER task, where the LSTM is utilized to extract the context semantic feature, and the CRF is used as the sequence tag decoder. The baseline model achieves the state of art at that time and is robust. LSTM can establish long distance dependence due to its network structure characteristics, but the prominent disadvantage of LSTM is that it cannot conduct parallel computing and make full use of GPU resources. In the current popular neural network, the Convolutional Neural

The associate editor coordinating the review of this manuscript and approving it for publication was Mu-Yen Chen.

Networks (CNN) has excellent performance in extracting local features, and can process text information in parallel. However, when extracting the high-order text features, the depth of the convolution layer needs to be increased, which increases the computing difficulty, and the ordinary CNN performs not well in extracting global information and temporal information.

In 2018, Bai et al. [3] proposed the Temporal Convolutional Network(TCN), whose core module includes three parts: causal convolution, dilated convolution and residual connection. The causal convolution module can effectively remain the sequence information. The dilated convolution still expand the receptive field when using less convolution layer. Therefore, TCN not only has the ability of parallel processing as CNN, also has the function to model long distance information as Recurrent Neural Network (RNN). We propose TCN as the backbone network of the model in this paper since the CNER task relies on text sequence information to extract context semantic features. To emphasize the local features of the text, we propose TCN network with dynamic convolution kernel. Meanwhile, the Chinese named entities are sparsely distributed, so we propose a multihead attention using dynamic scaling factors to enhancing the global key features of the text. The contributions of this paper are mainly included in the following two points:

1. To extract the sequence information and improve the model of parallel computing ability, we utilize the TCN as a baseline network. In order to enhance the local information key features, we utilize a set of dynamic attention convolution kernel in each layer of convolution, and use the attention score for different convolution kernel convolution results give different weights, weighted sum output after the final result.

2. Named entity are sparse distribution in the text, the correlation of the named entity internal characters is high, and the external is low. According to the characteristics, on the basis of attention mechanism, we propose to use dynamic scaling factor to calculate the correlation between characters.

The remaining chapters of this paper are arranged as follows. The second chapter introduces the related work, the third chapter expounds the relevant principles of the model proposed in this paper, the fourth chapter describes the experimental process and experimental results, and the fifth chapter summarizes our work.

II. RELATED WORK

Compared with English data sets, Chinese data sets have natural disadvantages. English text takes single word as the basic unit, which has natural advantages in word segmentation, while Chinese text is composed of characters. The recognition accuracy of Chinese named entities is closely related to word segmentation, and wrong word segmentation will also cause errors in downstream tasks. Therefore, how to accurately extract contextual semantic features is a key concern in the CNER task. In previous work [4], [5], [6], we found that character-based models perform better than word-based models, but character-based models utilize explicit word and word sequence information, which is often useful. Zhang and Yang [7] integrates the potential word information into the LSTM-CRF based on character features,using the lattice structure Lattice-LSTM to represent words from sentences. However, it is necessary to generate a dynamic grid structure when calling word information, so that the parallel computing ability of GPU cannot be fully utilized. Transformer utilizes the fully connected self-concern to model the remote dependencies in the sequence. To maintain position information, Transformer introduces a position representation for each token in the sequence. Inspired by this, Li et.al [8] proposed the input representation method called flat-lattice, and encoded information for each tokens, so that the lattice structure can be turned into a flat lattice structure through the formula. Ma et al. [9] proposed a CNER method based on Soft-lexicon encoding word information, encoding the information of character and word into a joint representation, which not only uses the boundary information of potential words, but also uses the semantic information of words. Wu et al. [10] proposed the MECT model, which takes the structural information of characters, words and Chinese characters as the multiembedding, and uses the two-stream model to integrate the multiembedding information. These studies have achieved good results in the field of CNER, but they need vocabulary information or other external resources, and the mobility is not strong [11]. Using deep neural network method to process the named entity recognition task has a more powerful ability to automatically capture the potential features, its recognition effect is superior to the traditional method, its representative models include bidirectional long and short-term memory network model and conditional random field fusion model (BiLSTM-CRF) [1], based on Transformer coding model such as BERT [12] and related optimization model such as Lattice-LSTM model [7]. And these models can extract word vector features containing context semantics, complete more accurate label prediction for each word. Huang et al. [13] used BiLSTM-CRF to complete the named entity recognition task with excellent results. Based on BiLSTM-CRF, Cao Chunping et al. [14] added convolution windows of different sizes to capture the boundary feature information between multiple words. Subsequently, the attention is widely used by researchers for its advantage of grasping the intrinsic association with data or traits. Li et al. [15] proposed methods based on the dynamic attention mechanism to enhance the model performance by stitching together the character vector of the original text information and the word vector of the domain information. Xu et al. [16] obtained the relationship between words in sentences through a self-attention mechanism, and proposed a NER model with a supervised multihead self-attention network. Chen et al. [17] uses the self-attention to receive input and continuously adjust the self-attention, and then proposes a lightweight low-resource cue-guided attention generation framework.

The self-attention mechanism used in the above studies has few parameters and low complexity, which greatly improves

the computational efficiency and captures the potential dependency weights, context and semantic associations of the data etc., but all of them suffer from the problem of not being able to learn the feature information of the textual sequences, and can only obtain the sequence information through the BiLSTM layer. The BiLSTM has been excellent in dealing with the problem of extracting the long distance dependency problem in several domains, but in the BiLSTM model, the state of the current moment depends on the state of the previous moment, which makes BiLSTM can not be computed in parallel, and brings the problem that the computation process becomes complicated. To improve the parallel computing ability of the model, the article [18] proposes to apply a CNN to the NER model, and the CNN performs well in local feature extraction, however, when extracting higher-order features of the text, such as temporal information, it is necessary to deepen the convolutional network layer. The article [19] uses a multihead attention to handle machine translation, opening the era of Transformer in NLP tasks. Unlike the previous Seq2Seq framework based on RNN, the attention mechanism is utilized instead of the RNN in Transformer in order to build the whole model, which reduces the dependence on external information, captures the internal relevance of the data and the features as well as the key features more efficiently, and it can capture the long distance dependency relationships, thus providing better information about the global contextual semantic features. Han and Li et al. proposed two CNER models around the Transformer method, among them, the article [20] analyzed the reason why the Transformer model is not suitable for the NER task from the perspective of mathematical formulas, and omitted the scaling factor of the named entities in the self-attention scores according to the sparse distribution of the named entities in the text in the NER task. In the article [8], in the input representation layer, the text of data sets is used to match potential words from the dictionary, the text of data sets and potential words are processed into tokens,and the tokens are encoded with the first and last positions, and then, the transformer method is used to model the relationship between the first and last positional information, which obtains the best results at that time. However, transformer is using a dense vector to model the whole data, which is computationally huge, and the results are not friendly to small sample data sets.

After being proposed, TCN has been widely used in the field of time series forecasting [21], [22], [23], and in recent years, it has started to be applied in NLP tasks such as text classification [24], text sentiment analysis [25],relationship extraction [26], and so on. Inspired by the fact that TCN and multihead attention mechanisms have been applied to other NLP tasks, we propose a model that combines dynamic temporal convolution with dynamic attention mechanisms. The TCN effectively preserves the sequence information of the text and avoids the impact of the loss of word order information on sentence meaning. In order to obtain more accurate local features of contextual semantics, we propose to improve the traditional dilated causal convolution by using a set of dynamic attention convolution kernel to process the input vectors and enhance the local contextual semantic features. In order to enhance the global key features and adapt to the scattered and sparse nature of Chinese named entities in text, after the dynamic TCN module, we propose a multihead attention using dynamic scaling factors to compute character vector relations.

III. MODEL

Our model is divided into three layers: input representation layer, contextual semantic feature extraction layer and feature decoding layer, and the overall framework is shown in Fig.1. The input representation layer uses Word2vec to convert text into 100-dimensional character vectors. In the contextual semantic feature extraction layer, we propose a dynamic TCN to extract contextual semantic features with character features and a dynamic attention to extract the global key feature information. And the feature decoding layer uses conditional random fields (CRF) to select the optimal entity label sequences in the output vectors of the feature extraction layer.

A. INPUT REPRESENTATION LAYER

In the layer, we adopt the Continuous Bag of Words model(CBOW) in the Word2vec method to process each character into a 100-dimensional embedding vector. Compared with continuous skip-gram model, CBOW predicts the center word by context words, which has higher computational accuracy and is more suitable for operations with larger number of texts. The calculation formula is shown in (1).

$$L = \sum_{w \in C} nP\left(w|Context\left(w\right)\right). \tag{1}$$

where $Context(w)$ denotes the context of the word w in the sentences, and w is any word in the corpus.

B. CONTEXTUAL SEMANTIC FEATURE EXTRACTION LAYER

The contextual semantic feature layer is based on the classical TCN as a baseline network, combined with a multihead self-attention mechanism module to extract key feature information on the input vectors.

1) DYNAMIC TEMPORAL CONVOLUTIONAL NETWORK

TCN is composed of a set of residual blocks containing one-dimensional dilated causal convolution, weight regularization, Relu, dropout and residual connection. The structure of the residual block is schematically shown in Fig.2. The causal convolution is applied to retain sequence information, and the dilated convolution relies on a larger receptive field to obtain longer history information. Therefore, dilated causal convolution neural network has the common advantages of RNN and CNN. Its structure is shown in Fig.3. Residual blocks are used to prevent gradient explosion, gradient vanishing, and gradient degradation problems associated with deeper networks. After each dilated convolutional computation, the parameters are normalized, then the Relu

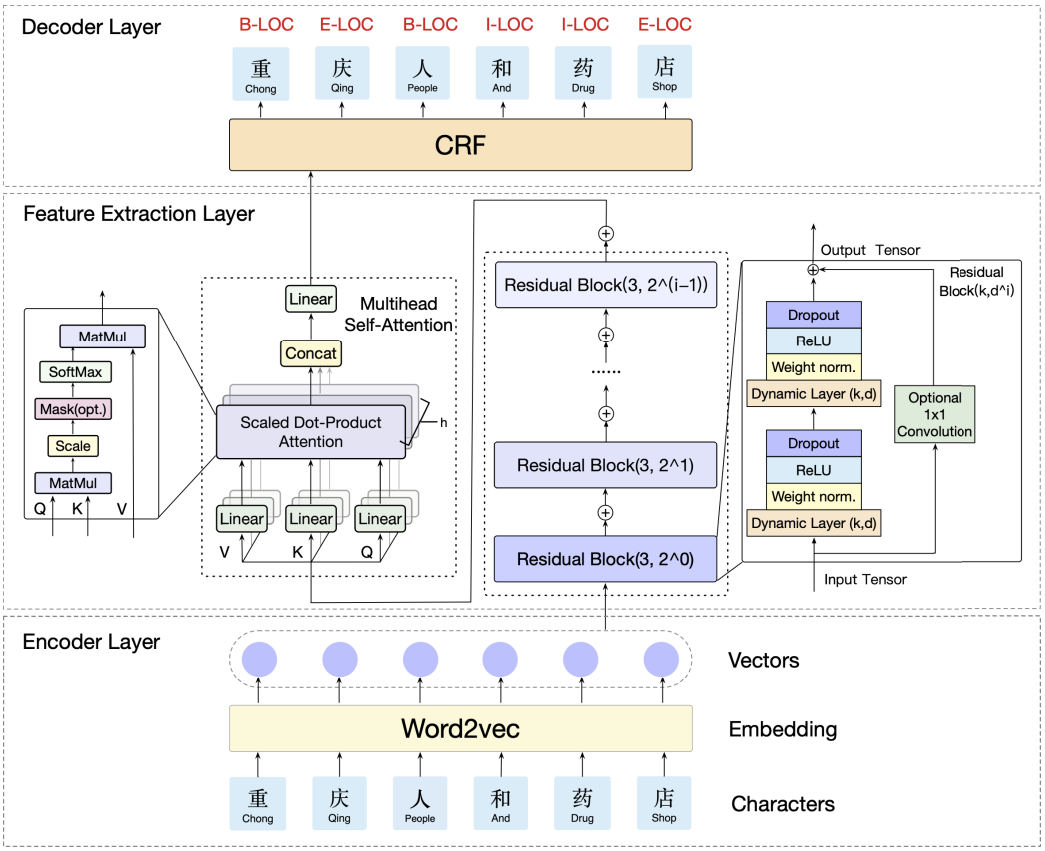


FIGURE 1. The overall framework of DTDM model.

function is used to complete the nonlinear computation, and finally the dropout operation is performed. The principle formula can be expressed as (2),(3),(4),(5):

$$S_i = Conv(h_i + K_j + b_i). \tag{2}$$

$$\{S_0, S_1, \ldots S_n\} = LayerNorm(\{S_0, S_1, \ldots S_n\}). \tag{3}$$

$$\{C_0, C_1, \ldots C_n\} = Relu(\{S_0, S_1, \ldots S_n\}). \tag{4}$$

$$\{D_0, D_1, \ldots D_n\} = Dropout(\{C_0, C_1, \ldots C_n\}). \tag{5}$$

where S_i denotes the result of embedding vector matrix computed by dilated causal convolution, h_i represents the ith character embedding vector, K_j denotes the convolution kernel of layer j, and b_i denotes the bias vector. $\{S_0, S_1, \ldots S_n\}$ denotes the normalized character eigenvectors. $\{C_0, C_1, \ldots C_n\}$ denotes the eigenvectors computed by nonlinear computation of the Relu function, and $\{D_0, D_1, \ldots, D_n\}$ denotes the eigenvectors computed by dropout. The character embedding vector h_i and the feature vector $\{D_0, D_1, \ldots D_n\}$ are connected by residuals to form a residual block output. The formula is shown as (6).

$$H = \{h_0, h_1, \ldots h_n\} + \{D_0, D_1, \ldots D_n\}. \tag{6}$$

For enhancing key local features in the text, a set of dynamic convolution kernel incorporating an attention mechanism is used instead of a one-dimensional convolution kernel, inspired by the article [27]. The weights of the dynamic convolution are generated by the attention mechanism generator, which consists of an average pooling layer, two fully connected layers, a Relu function layer, and a Softmax layer, and the attention mechanism generator generates a set of different attention scores based on the text, which are weighted and summed with the dynamic convolution kernel to obtain a weighted convolution kernel, which we use for localized text feature extraction. The TCN structure of the dynamic attention convolution kernel we use is shown in Fig.4. The calculation process of the dynamic convolution kernel is given in Table 1.

TABLE 1. Algorithm 1 description.

Algorithm 1:The Calculation Process of Dynamic Convolutional Kernel
Input: N input tensors Tensors S
Output: Dynamic Convolutional Kernel after Weighted Average
for i in range(0,n) then
$sc_i = Adapt_attention(S)$
$new_kernel+ = sc_i * kernel_i$
end for

2) DYNAMIC MULTIHEAD ATTENTION MECHANISM

Before introducing the multihead attention mechanism, we introduce the self-attention mechanism, which is formulated as follows:

$$Att(A, V) = softmax(A)V. \tag{7}$$

$$A_{ij} = \left(\frac{Q_i K_j^T}{\sqrt{d_{\text{head}}}}\right). \tag{8}$$

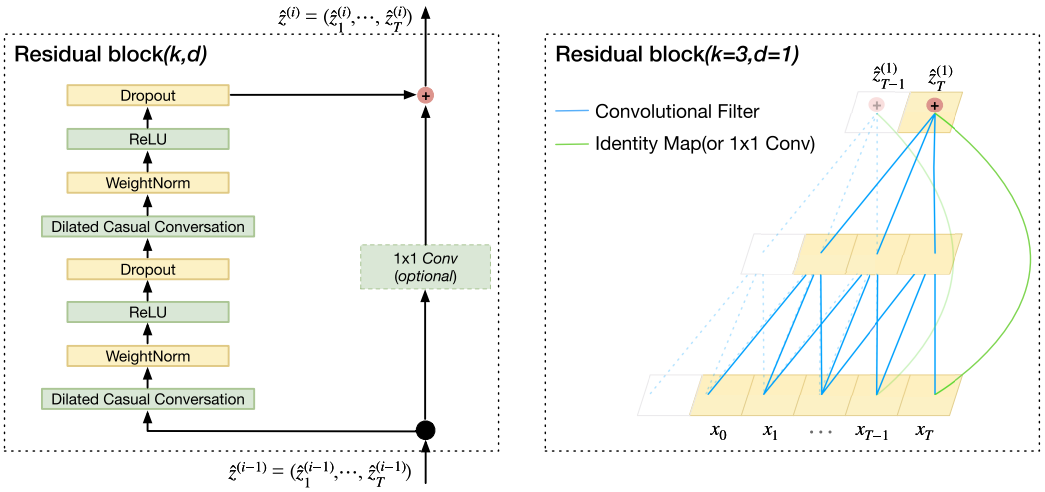


FIGURE 2. The residual network in TCN.

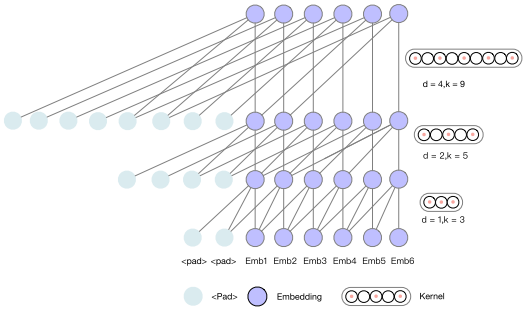


FIGURE 3. Dilated casual convolutional network.

$$[Q, K, V] = E_x \left[W_q, W_k, W_v\right]. \tag{9}$$

where Att $(\cdot)$ represents the attention score, A_{ij} represents the correlation score between the ith and jth vectors. Q, K, and V are obtained by the input vector E_x and query weight matrix W_q, key value weight matrix W_k, value weight matrix W_v is obtained by multiplying them separately.

The input information is divided into h groups by the multihead self-attention, and projected in each group of inputs into Q (Query), K (Key), V (Value) spaces for linear transformation. Then, parallel attention pooling operations are performed on the transformed queries, keys, and values in each group. Finally, the outputs of h attention heads pooling are concatenated together, and are transformed to final output by another learnable linear projection W_0 and the calculation process is shown in (10) (11).

$$h_i = f\left(W_i^q q, W_i^k k, W_i^v v,\right). \tag{10}$$

$$Multihead(h) = W_o \left[h_1, \ldots, h_h\right]^T. \tag{11}$$

where W_i^q, W_i^k, W_i^v, W_o are learnable parameters. The function f$(\cdot)$ representing attention pooling can be additive attention and scaled ''dot-product'' attention.

There are a small number of named entities in the sentence. Removing the scaling can make attention sharper and achieve a more focused attention distribution. This method achieved good results, but ignores the original intention of introducing a scaling factor. When the value of dot-product is too large during the calculation process, the gradient disappears to 0, making parameter updates difficult [28]. Therefore, we propose a dynamic scaling factor based on the above literature. This method can adjust the attention weight values of entities and irrelevant words according to the output of the hidden layer, alleviate the interference of irrelevant words on the model, and obtain a more suitable attention distribution for CNER. And the method alleviate the problem caused by excessive inner product, improving the effectiveness of NER. Inspired by the ELU function, in order to adapt the attention score calculation, the following function is constructed:

$$G(x) = \begin{cases} x + 1, & x \geq 0 \\ e^x. & x < 0 \end{cases} \tag{12}$$

where x is the last layer input.Its gradient function formula $Grad(x)$ such as (13). Images of them are shown in Fig.5.

$$G(x) = \begin{cases} 1, & x \geq 0 \\ e^x. & x < 0 \end{cases} \tag{13}$$

From (12)(13), the value domain of $G(x)$ is $(0, \infty)$,and the value domain of its gradient function $Grad(x)$is $(0, 1]$. $G(x)$ is continuously differentiable at all points. It will not suffer from gradient explosion or disappearance problem as a nonsaturable activation function. However, its calculation speed is slow since it is nonlinear in the negative number.

The formula for the dynamic scaling factor in this article is as follows:

$$\eta_i^l = min\left(G\left(W\left(h_i^{l-1}\right)^T\right), \sqrt{d_{head}}\right) + 1. \tag{14}$$

where η_i^l represents the dynamic scaling factor of the ith character in the lth layer, W is a learnable parameter, and h_i^{l-1} represents the output of layer $l-1$. The calculation process of the dynamic scaling factor is shown in Table 2.

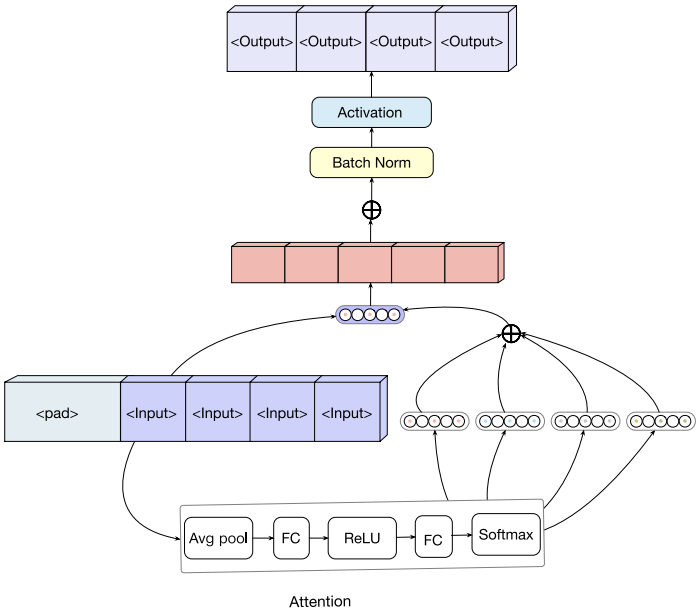


FIGURE 4. The detail of dynamic TCN convolutional kernel.

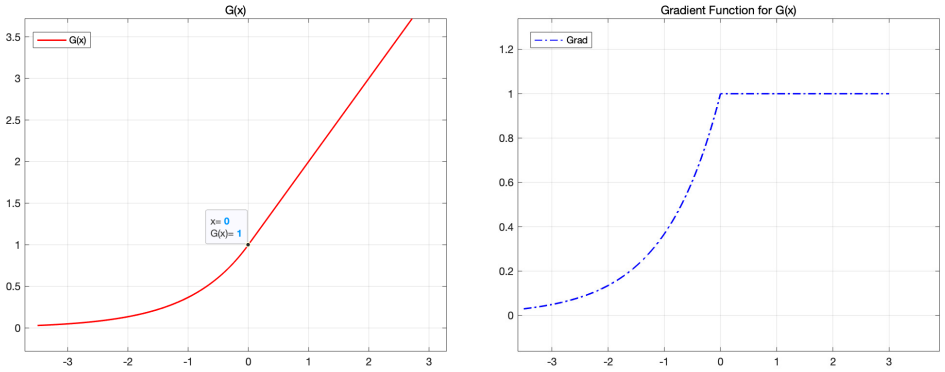


FIGURE 5. The left is the image of G(x); the right is the image of Grad(x).

TABLE 2. Algorithm 2 description.

Algorithm 2:The calculation process of the dynamic scaling factor
Input: The hidden state of the $l-1$ layer's output
Output: Dynamic scaling factor for l layer
for vector h in H_{l-1} do
$temp = wh^T$
if $temp \geq 0$ then
$G = temp + 1$
else
$G = min(G, \sqrt{d_{head}}) + 1$
$dy.append(\eta)$
end for

C. SEMANTIC FEATURE DECODING LAYER

The CRF can effectively solve such problems by learning the adjacency between tags. Therefore, this article uses the CRF to model the dependency relationship between sequence labels in the semantic feature decoding layer, and obtains the optimal label sequence. The CRF can fully utilize the contextual information of entity labels and learn the transfer relationship between labels by training a transfer probability matrix, solving the problem of label bias. From a global modeling perspective, The CRF uses logarithmic maximum likelihood estimation to calculate the loss function to maximize the probability of the correct sequence, and finally selects a path with the highest score as the tag sequence, such as sentence $S_i = \{C_1, C_2, \ldots C_l\}$. After passing through the feature extraction layer, the extracted feature matrix is $T_i = \{t_1, t_2, \ldots t_n\}$ to obtain the predicted sequence $D_i = \{d_1, d_2, \ldots d_n\}$, the calculation of its fractional function is shown in (15).

$$S(X, Y) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} P_{i, y_i}. \tag{15}$$

where $A_{y_i, y_{i+1}}$ represents the probability of label y_i transitioning to label y_{i+1}, P_{i,y_i} represents the equal probability of the i word being predicted as the j label, $S(X, Y)$ represents the probability score of the input sentence sequence X being marked as Y. The probability generated by the predicted sequence Y is shown in (16):

$$P(X|Y) = \frac{e^{S(X,Y)}}{\sum_{\tilde{Y} \in Y_X} S\left(X, \tilde{Y}\right)}. \tag{16}$$

TABLE 3. Labels of named entity types in the Weibo dataset.

Named Entity Types	Labels
PER	PER.NAM
	PER.NOM
LOC	LOC.NAM
	LOC.NOM
ORG	ORG.NAM
	ORG.NOM
GPE	GPE.NAM

where $\tilde{Y}$ represents the real annotation sequence; Y_X represents all possible annotation sequences. When decoding, we use the Wibit algorithm to find the Y^* with the highest score among all Y, as shown in (17):

$$Y^* = \underset{\tilde{Y} \in Y_X}{\operatorname{argmax}} S(X, \tilde{Y}). \tag{17}$$

where Y^*represents the global optimal annotation sequence.

IV. EXPERIMENTS

A. EXPERIMENTAL SETUP

This experiment uses Weibo dataset to validate our model. Weibo dataset is a information corpus collected from Sina Weibo. The dataset includes Name, Location, Administrative area and Organization. which are annotated with PER.NAM, LOC.NAM, ORG.NAM, and GPE.NAM, respectively. Meanwhile, pronouns for personal names, generic references for addresses, and generic and generic references for organizations end with NOM. The label of named entity types in the dataset is shown in the Table 3. The training dataset includes 1350 sentences, the development dataset includes 270 sentences, and the test dataset includes 270 sentences.

The dataset is labeled with BIO labeling method, which classifies the entities in the dataset into three kinds of sequence labeling according to their lengths and categories.The B-X is for the beginning of entity X, the I-X for the middle or the end of entity X, and O for not belonging to any entity type. The experiments use Precision (P), Recall (R), and F1 value as evaluation metrics [29], calculated as in Eqs. (16),(17),(18):

$$P = \frac{TP}{TP + FP}. \tag{18}$$

$$R = \frac{TP}{TP + FN}. \tag{19}$$

$$F1 = \frac{2PR}{P + R}. \tag{20}$$

True Positive(TP): Comments that were initially categorized as positive and were projected to be positive by the classifier.

False Positive(FP): Comments that were initially categorized as positive but were projected by the classifier to be negative.

False Negative(FN): Comments that were categorized as negative but were predicted as positive by the classifier.

B. RESULTS AND ANALYSIS

The F1 value of our model on the Weibo dataset reaches 89.24%, and the experimental results validate the effectiveness of the proposed model.

In order to demonstrate the effectiveness of the DY-MHA-DY-TCN model proposed in this paper, we have chosen to compare the classical model in terms of three evaluation indexes, namely Precision, Recall, and F1 value. The comparison models are introduced as follows:

Lattice-LSTM: Zhang et al. [7] proposed using Lattice based LSTM to represent lexicon words in sentences, thereby embedding potential vocabulary information into character based on LSTM-CRF.

TENER: YAN et al. [20] proposed an attention mechanism with position perception to improve the Transformer model, which can capture the position and direction information of words, and model contextual information at the word and character levels.

Flat-Lattice: Li et al. [8] proposed to simultaneously encode the position information of characters and potential words, calculate the difference between the first and last encoding, convert Lattice-LSTM into a flat structure, use Transformer to learn the first and last relationship information, and use conditional random field to predict the output of sequence tags.

SoftLexicon: Ma et al. [9] proposed encoding dictionary information into vector representations to avoid complex model structures, improve computational speed, and enhance compatibility between dictionary structures and other neural network models.

LEBERT: Liu et al. [30] embedded the Lexicon adapter layer into the transformer layer of BERT to integrate dictionary information, enabling sentences to be encoded as character words pairs, resulting in both character and dictionary features in the output of LEBERT.

TBAC: Liao et al. [11] proposed to improve the Transformer encoder by using the relative position encoding and modifying the attention calculation formula to provide global semantic information, and to capture the direction information using BiLSTM. On this basis, combined with the attention mechanism, the weight is dynamically adjusted, and the global semantic information and direction information are deeply fused to obtain richer context features.

NFLAT: Wu et al. [31] proposed an Inter Former module with multi-head inter attentions to construct a none flat lattice model, which can simultaneously model character and vocabulary sequences of different lengths and reduce some redundant calculations.

BSNER: Zhu et al. [32] proposed the use of boundary smoothing as a span-based NER model regularization technique to redistribute the probability of an entity from the labeled span to the span around the entity, which can effectively alleviate the overconfidence problem that neural network models are prone to encounter and bring about smoother model predictions.

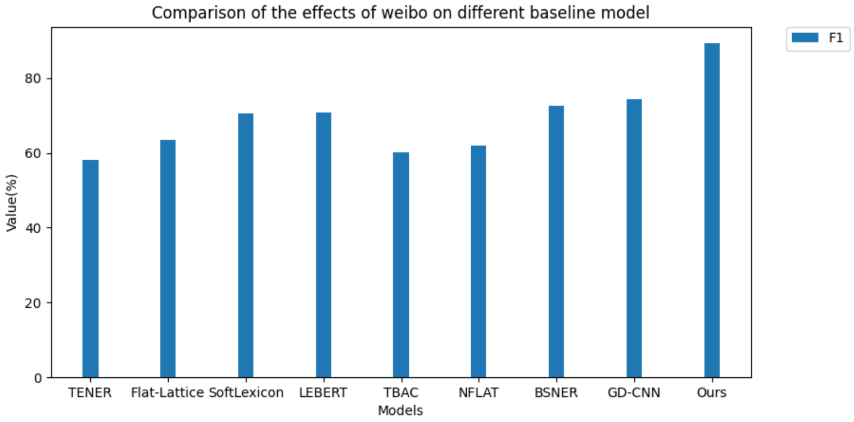


FIGURE 6. Comparison results of Weibo dataset on different models.

TABLE 4. Comparison results of Weibo dataset on different models.

	Precision(%)	Recall(%)	F1(%)
Lattice-LSTM(2018)	—	—	58.79
TENER(2019)	—	—	58.17
Flat-Lattice(2020)	—	—	63.42
SoftLexicon(2020)	—	—	70.50
LEBERT(2021)	—	—	70.75
TBAC(2022)	69.60	53.11	60.24
NFLAT(2022)	—	—	61.94
BSNER(2022)	70.16	75.36	72.66
GD-CNN(2022)	77.91	70.97	74.28
DY-MHA-DY-TCN(ours)	87.24	92.74	89.24

GD-CNN: Yang et al. [33] uses the pretraining language model Ro-BERTa-wwm based on the whole word masking technology to represent the text as a character level embedding vector, capture the depth context semantic information, improve the dilated convolution neural network with the gating mechanism and residual structure to reduce the risk of gradient disappearance, and capture the temporal and spatial characteristics of the text respectively through BiLSTM and Gated Dilated Convolution Network. Then, a bilinear multihead attention mechanism is used to dynamically fuse multidimensional text features, and the CRF layer is used to constrain the results to obtain the optimal label sequence.

The comparison results are shown in Table 4 and Fig.6.

C. ABLATION EXPERIMENT

To verify the effectiveness of the proposed model, we conducted a set of ablation experiments on the basis of the proposed model. TCN+MHA represents using traditional TCN and traditional multihead attention mechanism as encoders; DY-TCN+MHA represents using dynamic TCN and traditional multihead attention mechanism as encoders, TCN+DY-MHA represents using traditional TCN and dynamic attention mechanism as encoders. The results are as shown in Table 5.

The Precision of the proposed DY-MHA-DY-TCN model is 0.47%, -0.81%, -0.43% higher than that of TCN+MHA, DY-TCN+MHA and TCN+DY-MHA respectively. The Recall is 0.72%, 0.08%, 0.25% higher than that of TCN+MHA, DY-TCN+MHA, TCN+DY-MHA respectively. The F1 value is 0.31%, 0.06%, 0.11% higher than that of TCN+MHA, DY-TCN+MHA, TCN+DY-MHA respectively. The results show that our proposed model is effective on a small data sets in NER.

TABLE 5. The results of ablation experiment.

	Precision(%)	Recall(%)	F1(%)
TCN+MHA	86.77	92.02	88.93
DY-TCN+MHA	88.05	92.62	89.18
TCN+DY-MHA	87.67	92.49	89.13
DY-TCN-DY-MHA(ours)	87.24	92.74	89.24

V. CONCLUSION

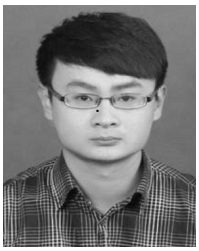
Our proposed model enhances the contextual semantic feature information from both local and global features. A set of dynamic convolutional kernel is used in the TCN layer for better extracting key local features, while a dynamic scaling factor formula is proposed in our study to compute the correlation between characters in conjunction with a multihead attention mechanism to obtain global contextual semantic feature information. The experimental results show that the experimental results of our proposed model in the three evaluation metrics of Precision, Recall, and F1 value are all greatly improved than the latest related studies, although we only use the Word2vec encoding method in the input representation layer without using external resources such as external dictionaries,which proves that our proposed model has better effectiveness in extracting contextual semantic feature information in CNER task. In our future work, we apply the proposed model to more specific application scenarios.

REFERENCES

[1] K. Xu, Z. Yang, P. Kang, Q. Wang, and W. Liu, "Document-level attention-based BiLSTM-CRF incorporating disease dictionary for disease named entity recognition," *Comput. Biol. Med.*, vol. 108, pp. 122–132, May 2019.

[2] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," 2016, *arXiv:1603.01360*.

[3] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.
[4] J. He and H. Wang, "Chinese named entity recognition and word segmentation based on character," in *Proc. 6th SIGHAN Workshop Chin. Lang. Process.*, 2008, pp. 128–132.
[5] Z. Liu, C. Zhu, and T. Zhao, "Chinese named entity recognition with a sequence labeling approach: Based on characters, or based on words?" in *Proc. Int. Conf. Intell. Comput.*, Changsha, China, 2010, pp. 634–640.
[6] H. Li, M. Hagiwara, Q. Li, and H. Ji, "Comparison of the impact of word segmentation on name tagging for Chinese and Japanese," *LREC*, vol. 2014, pp. 2532–2536, 2014.
[7] Y. Zhang and J. Yang, "Chinese NER using lattice LSTM," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2018, pp. 1–11.
[8] X. Li, H. Yan, X. Qiu, and X. Huang, "FLAT: Chinese NER using flat-lattice transformer," 2020, *arXiv:2004.11795*.
[9] R. Ma, M. Peng, Q. Zhang, and X. Huang, "Simplify the usage of lexicon in Chinese NER," 2019, *arXiv:1908.05969*.
[10] S. Wu, X. Song, and Z. Feng, "MECT: Multi-metadata embedding based cross-transformer for Chinese named entity recognition," 2021, *arXiv:2107.05418*.
[11] L. Liao and S. Xie, "Chinese named entity recognition based on feature fusion of attention mechanism," *Comput. Eng.*, vol. 49, no. 4, pp. 256–262, 2023.
[12] F. Souza, R. Nogueira, and R. Lotufo, "Portuguese named entity recognition using BERT-CRF," 2019, *arXiv:1909.10649*.
[13] Z. Huang, X. Wei, and Y. Kai, "Bidirectional LSTM-CRF models for sequence tagging," 2015, *arXiv:1508.01991*.
[14] C. P. Cao and P. J. Guan, "Clinical text named entity recognition based on E-CNN and BLSTM-CRF," *Appl. Res. Comput.*, vol. 36, no. 12, p. 4, 2019.
[15] Y. Li, G. Du, Y. Xiang, S. Li, L. Ma, D. Shao, X. Wang, and H. Chen, "Towards Chinese clinical named entity recognition by dynamic embedding using domain-specific knowledge," *J. Biomed. Informat.*, vol. 106, Jun. 2020, Art. no. 103435.
[16] Y. Xu, H. Huang, C. Feng, and Y. Hu, "A supervised multi-head self-attention network for nested named entity recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 16, pp. 14185–14193.
[17] X. Chen, L. Li, S. Deng, C. Tan, C. Xu, F. Huang, L. Si, H. Chen, and N. Zhang, "LightNER: A lightweight tuning paradigm for low-resource NER via pluggable prompting," 2021, *arXiv:2109.00720*.
[18] E. Strubell, P. Verga, D. Belanger, and A. McCallum, "Fast and accurate entity recognition with iterated dilated convolutions," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 1–13.
[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[20] H. Yan, B. Deng, X. Li, and X. Qiu, "TENER: Adapting transformer encoder for named entity recognition," 2019, *arXiv:1911.04474*.
[21] L. Qiu, L. Jin, and L. Chai, "Network traffic prediction based on spatio-temporal graph convolutional network," in *Proc. 42nd Chin. Control Conf. (CCC)*, Kaifeng, China, Jul. 2023, pp. 144–155.
[22] J. F. Torres, M. Jiménez-Navarro, F. Martínez-Álvarez, and A. Troncoso, "Electricity consumption time series forecasting using temporal convolutional networks," in *Proc. 19th Conf. Spanish Assoc. Artif. Intell.*, Málaga, Spain, Sep. 2021, pp. 216–225.
[23] Y.-F. Zhang, P. J. Thorburn, and P. Fitch, "Multi-task temporal convolutional network for predicting water quality sensor data," in *Proc. Int. Conf. Neural Inf. Process.*, Sydney, NSW, Australia, Dec. 2019, pp. 122–130.
[24] Y. Liu, P. Li, and X. Hu, "Combining context-relevant features with multi-stage attention network for short text classification," *Comput. Speech Lang.*, vol. 71, Jan. 2022, Art. no. 101268.
[25] D. Cao, Y. Huang, H. Li, X. Zhao, Q. Zhao, and Y. Fu, "Text sentiment classification based on LSTM-TCN hybrid model and attention mechanism," in *Proc. 4th Int. Conf. Comput. Sci. Appl. Eng.*, Oct. 2020, pp. 1–5.
[26] Y. Shi, Y. Xiao, P. Quan, M. Lei, and L. Niu, "Document-level relation extraction via graph transformer networks and temporal convolutional networks," *Pattern Recognit. Lett.*, vol. 149, pp. 150–156, Sep. 2021.
[27] Y. Chen, X. Dai, M. Liu, D. Chen, L. Yuan, and Z. Liu, "Dynamic convolution: Attention over convolution kernels," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11027–11036.
[28] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-XL: Attentive language models beyond a fixed-length context," 2019, *arXiv:1901.02860*.
[29] W. Jing, X. Song, D. Di, and H. Song, "GeoGAT: Graph model based on attention mechanism for geographic text classification," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 20, no. 5, pp. 1–18, Sep. 2021.
[30] W. Liu, X. Fu, Y. Zhang, and W. Xiao, "Lexicon enhanced Chinese sequence labeling using BERT adapter," 2021, *arXiv:2105.07148*.
[31] S. Wu, X. Song, Z. Feng, and X.-J. Wu, "NFLAT: Non-flat-lattice transformer for Chinese named entity recognition," 2022, *arXiv:2205.05832*.
[32] E. Zhu and J. Li, "Boundary smoothing for named entity recognition," 2022, *arXiv:2204.12031*.
[33] C. Yang and S. Liao, "Chinese named entity recognition based on gated-dilated convolution feature fusion," *Comput. Eng.*, vol. 49, no. 8, pp. 85–95, 2023.



YUAN HUANG was born in Hebei, China, in 1987. He received the bachelor's and master's degrees from the Hebei University of Engineering, in 2010 and 2013, respectively, and the Ph.D. degree from Yanshan University, in 2017. Since 2017, he has been a Teacher with the School of Information and Electrical Engineering, Hebei University of Engineering. He has published 11 articles. His research interests include data mining and machine learning.

YANXIA LI was born in Shandong, China, in 1995. She received the bachelor's degree from Shandong Technology and Business University, in 2017. She is currently pursuing the master's degree with the School of Information and Electrical Engineering, Hebei University of Engineering. Her research interests include data mining, deep learning, and NLP.

XIAOYU ZHANG was born in Shanxi, China, in 1998. He received the bachelor's degree from the Business College, Shanxi University, in 2021. He is currently pursuing the master's degree with the School of Information and Electrical Engineering, Hebei University of Engineering. His research interests include data mining and machine learning.