Short text classification based on bidirectional TCN and attention mechanism

Ying Zuo, Lifen Jiang*, Huazhi Sun, Chunmei Ma, Yan Liang, Shuaibao Nie, Yongheng Zhou

College of Computer and Information Engineering, Tianjin Normal University, Tianjin, 300387, China

kathhhh@126.com, jianglifen@tjnu.edu.cn

Abstract. The context-related semantic information of the text in the traditional short text classification algorithms are not fully captured, a text classification model based on bidirectional temporal convolutional network and attention mechanism (BTCA) was proposed. Multi-layer dilated convolution was used to increase the receptive field and better capture long distance dependent information. At the same time, attention mechanism was used to increase the attention to the local key feature in the text, and the bidirectional temporal convolution network was used to extract contextual multi-scale semantic information to enrich semantics, the problem of sparse short text features was solved to a certain extent, and text classification effect was improved. The public corpus of the THUCNews was used to conduct a comparative experiment. It is pointed out that effect of short text classification is improved by using BTCA model, with an accuracy rate of 91.47%, which is better than commonly used models.

1. Introduction

How to classify and manage text quickly and effectively is a hot issue in the field of natural language processing (NLP). Short text classification is one of the important methods to understand short texts. Compared with paragraphs, short texts have fewer words, which tends to cause data sparseness and insufficient context information. Therefore, short text classification poses huge challenges.

At present, the mainstream research methods mostly use the extraction of local key feature and then classify them based on them. In the field of deep learning, Kim et al. [1] applied Convolutional Neural Network (CNN) to text classification tasks, using multiple convolution kernels of different sizes to extract key information in sentences, which can capture the local relevance of text. Zhang et al. [2] tried to use character-level convolutional networks for text classification. Hao et al. [3] used a Bi-LSTM model for sequence modeling, and Wang et al. [4] introduced residual network into RNN to enable the model to process longer sequences. Guo et al. [5] proposed an attention-based hybrid neural network CRAN based on CNN-RNN, which effectively combines CNN and RNN with the help of attention mechanism. Vaswani et al. [6] applied the attention mechanism to NLP, and Letarte et al. [7] applied the attention mechanism to text classification, which proved the effectiveness of the attention mechanism. The Temporal Convolutional Network (TCN) proposed by Bai et al. [8] is a network structure that can process sequence and has achieved good classification results. But in dealing with the classification of the word "apple" as shown in Table 1, if there is not enough context, it may lead to incorrect classification. The real meaning of “apple” can only be correctly recognized from overall view instead of narrow window. If the relevant information such as "fresh fruit" and "Huawei" in the context is not available, it is difficult to obtain the correct classification effect on a local scale.
Table 1: Classification examples for Society and Technology

| Case1: | Apples are not selling smoothly, fresh fruits are successively listed. |
| Case2: | Apple’s sales data released, and Huawei head-to-head! |

In order to solve these problems, we propose a Bidirectional temporal convolutional network based on the attention mechanism—BTCA model. First, model the sequence to fully capture the long distance contextual information, and then extract the local feature and classify short text.

2. BTCA Model

We propose the BTCA model, and its overall architecture is shown in Fig. 1.

Fig. 1 The overall framework of BTCA model

The BTCA contains four modules: the input embedding module generates word representation $X$ at the word level of the short text; the context information module models the short text through a Bi-TCN and generates short text representation $H$; attention layer extract local key information from $H$; the classifier module obtains the probability of each class label through the activation function.

2.1. Input Embedding

This module utilizes Word2vec method in gensim machine learning library to map the preprocessed word sequence to word vector. For a text, $W = \{w_1, w_2, \ldots, w_m\}$ is its word sequence, and word vector can be expressed as:

$$X = \{x_1, \; x_2, \; \ldots, \; x_m\} \in \mathbb{R}^{d \times m} \quad (1)$$

Where $x_i (1 \ll i \ll 10)$ represents the word vector of the i-th word, $d$ represents the dimension of the word vector, and $m$ represents the length of the short text.

2.2. Context information

In order to capture contextual information, we models the interaction relationship between words on the basis of input embedding module, uses Bi-TCN to obtain contextual information. $H$ is the representation of short text by concatenating the output from two directions.

TCN is a one-dimensional convolutional neural network used for sequence modeling tasks. It performs better than LSTM on various tasks and datasets without increasing parameters. TCN uses dilated convolution, for a one-dimensional input sequence $X \in \mathbb{R}$, the dilated convolution operation $F$ is described as follows:

$$F(s) = \sum_{i=0}^{k-1} f(i) \cdot X_{s-d \cdot i} \quad (2)$$
Where \( d \) represents the dilation rate, \( k \) represents the size of the convolution kernel, and \( X_{s=d^{-1}} \) represents the output of the previous layer. In this paper, we set up four layers of TCN, the dilation rate \( d \) is set to 1, 2, 3, so that the model accuracy can reach the best, as shown in Fig.2.

As the network depth increases, richer context features and more context-dependent information can be captured. However, deeper neural networks are more difficult to train, and increasing depth will also bring about problems such as vanishing gradients. For this reason, we introduces the residual block shown in Fig.3.

In the residual block, the output of each residual block is the nonlinear transformation of the input and the input. The input is mapped equally without generating additional parameters. The input nonlinear transformation introduces two layers of dilated casual convolution, and then use ReLU activation function after the convolution layer. It can expands the receptive field.

2.3. Attention layer

Bahdanau D et al. [9] introduced the attention mechanism to the field of NLP for the first time in the literature. In order to extract the key information of the short text, this paper uses the attention mechanism to increase the attention to the key local information. In the training process, the key local information is learned by updating the parameter matrix, ignoring redundant information, and the attention vector is calculated as follows:

\[
U = \tanh (W_u H + b_u) \quad (3)
\]

\[
a = \text{softmax}(U) \quad (4)
\]
\[ c = Hα^T \]  \hspace{1cm} (5)

Where \( W_w \) is a learnable parameter, \( b_w \) is a bias vector, and \( H \) is matrix composed of previous layer output vector \([h_1, h_2, \ldots, h_m]\), where \( m \) is the length of the sentence. \( H \) is passed through MLP and softmax function, then attention weight matrix is obtained. The sentence representation \( c \) is composed of the weighted sum of these output vectors.

### 2.4. classifier

The output of the network is the probability distribution of category labels. We use \( p(y | c, \phi) \) to represent the probability that the short text is of type \( y \), where \( \phi \) is a parameter in the network. In this paper, the text representation vector is input into the softmax classifier to predict the probability of each category, and cross entropy is used as the loss function:

\[ \hat{y} = \text{softmax}(W_cM + b_c) \]  \hspace{1cm} (6)

\[ L(y, \hat{y}) = \sum_i y_i \log \hat{y}_i \]  \hspace{1cm} (7)

Where \( y_i \) is the predicted probability, \( y_i \) is the actual probability of class \( i \), \( W_c \) is a learnable parameter, and \( b_c \) a bias.

### 3. Experiments

#### 3.1. Dataset

We use THUCNews dataset for experiments. The dataset is filtered and generated based on the historical data of the Sina News RSS subscription channel from 2005 to 2011, containing 740,000 News documents. We randomly extracts 70% of the dataset as the training set and 30% as the test set. Each news contains headlines, documents and topics (for example, 14 categories such as finance, education, sports, etc.). We obtain experimental data for the title as a short text, and the subject as a label. The title is generally between 10 and 20 words.

#### 3.2. Experimental Settings

We use experiments and combined experience to adjust parameters. The adjusted parameters include the number of Bi-TCN layers, the number of hidden layer nodes, etc., so that the model has the best accuracy in the dataset.

1. The number of layers of Bi-TCN

   In the process of text classification training using BTCA, the number of layers of Bi-TCN is verified experimentally, as shown in Table 2. The accuracy of the model increases first and then decreases as the number of Bi-TCN layers increases. In order to maintain good performance of the model, a four-layer Bi-TCN is used.

   | The number of layers of Bi-TCN | 3-level | 4-level | 5-level |
|-----------------------------|--------|--------|--------|
| Accuracy                    | 91.26% | 91.47% | 91.39% |

2. Bi-TCN hidden layer node number

   We conduct experiments on different Bi-TCN hidden layer nodes and observe the accuracy of the model. The experimental results are shown in Table 3. It is more appropriate to set the number of Bi-TCN hidden layer nodes to 300.

   | Bi-TCN hidden layer node number | 150 | 300 | 450 |
|-------------------------------|-----|-----|-----|
| Accuracy                      | 91.2% | 91.47% | 91.42% |

The settings of other parameters are shown in Table 4.
3.3. Experimental Results
In order to fully verify the effectiveness of the model, we conducted comparative experiments between the BTCA model and baseline models.

(1) Comparative experiment with CNN
Fig.4 shows the comparison results of BTCA and CNN models. As shown in Fig.4 (a), loss value of BTCA converges quickly. During the training process of the model, when the training reaches the 6th epoch, the loss tends to be stable. At this time, reduce the learning rate and continue training, the loss value of CNN is always higher than BTCA. It can be seen from Fig.4 (b) and Fig.4 (c) that the two evaluation indicators of BTCA, F1 and Recall, are higher than the CNN model. For the news classification example mentioned in the introduction of this article, CNN can extract key words for classification, such as "apple", but this word has multiple meanings and may be misclassified by CNN, while the BTCA model can capture the sequence after modeling the long distance information below, such as "fresh fruit" and "Huawei", is important for the model to obtain the correct meaning of "apple".

(2) Comparing experiments with baseline models: CNN, LSTM[10], TCN, DPCNN[11], the experimental results are shown in Table 3.

| Models   | Accuracy(%) |
|----------|-------------|
| CNN      | 91.09       |
| LSTM     | 90.82       |
| DPCNN    | 91.24       |
| TCN      | 91.22       |
| BTCA (our model) | **91.47** |

It can be seen from the experimental results in Table 3 that under the same experimental environment, the accuracy of the BTCA model proposed in this paper on the THUCNews dataset is 91.47%, which is better than the existing deep neural network models. The performance of LSTM model on this dataset is not as good as that of BTCA. LSTM model does not consider local key information, and LSTM model also has the problem of vanishing gradient. The performance of DPCNN model on the THUCNews dataset is also lower than the BTCA model. TCN model adopts a

### Table 4: The accuracy of each model on the THUCNews dataset

| Parameters                  | Value |
|-----------------------------|-------|
| Embedding Size              | 300   |
| Batch Size                  | 64    |
| Learning Rate               | 0.001 |
| Bi-TCN Kernel Size          | 3     |
| Bi-TCN Dropout              | 0.5   |
unidirectional temporal convolutional network, it cannot capture the dependence of the following information, and has no advantage compared with the BTCA model.

4. Conclusion
The BTCA model proposed in this paper comprehensively considers long distance dependence and local key information. In BTCA model, multi-layer dilated convolution is used to increase the receptive field to better capture the long distance dependence; the attention mechanism is used to increase the attention to the local key information in the short text and optimize the short text representation. In BTCA model, Bi-TCN is used to extract contextual multi-scale information, with richer semantics. The experimental results show that the BTCA model proposed in this paper has advantages compared with similar models, and it can complete the short text classification task better.

Acknowledgments
This work was financially supported by the National Natural Science Foundation (61702370); Tianjin Natural Science Foundation (18JCYBJJC85900, 18JQNJJC70200); Scientific Research Project of Tianjin Education Commission (JW1702); Tianjin Normal University 131 three-level candidates (043/153505QS20).

References
[1] Kim Y. Convolutional Neural Networks for Sentence Classification [J]. Association for the Computer Linguistics: 2014, 15(6): 1746-1751.
[2] Zhang Xiang, Zhao Junbo, LeCun Y. Character-level Convolutional Networks for Text Classification [C]. Neural Information Processing Systems. NIPS, 2015: 649-657.
[3] Hao Yanchao, Zhang Yuanzhe, et al. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge [C]. In Meeting of the Association for Computational Linguistics. Vancouver: ACL, 2017: 221–231.
[4] Wang Y, Tian F. Recurrent residual learning for Sequence classification [C]. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Austin: EMNLP, 2016: 938-943.
[5] Guo Long, Zhang Dongxiang, Lei Wang, et al. CRAN: A Hybrid CNN-RNN Attention-Based Model for Text Classification [C]. In Proceedings of ER 2018. Xi'an: ER, 2018: 571-585.
[6] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C]. Proceedings of 2017 Advances in neural information processing systems. California: NIPS, 2017: 5998-6008.
[7] Letarte G, et al. Importance of self-attention for sentiment analysis [C]. Proceedings of the 2018 EMNLP Workshop Blackbox-NLP. Brussels: EMNLP. 2018: 267-275.
[8] Bai Shaojie, Kolter J Z, Koltun V. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling [EB/OL]. [2019-09-01]. https://ui.adsabs.harvard.edu/abs/2018arXiv180301271B.
[9] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate [C]. ICLR. 2015.
[10] Duyu Tang, Bing Qin, and Ting Liu. Document modeling with gated recurrent neural network for sentiment classification [C]. Proceedings of the 2015 conference on empirical methods in natural language processing, Lisbon: EMNLP, 2015: 1422–1432.
[11] Rie Johnson, Tong Zhang. Deep Pyramid Convolutional Neural Networks for Text Categorization [C]. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver: ACL, 2017: 562-570.