Deep Pyramid Convolutional Neural Network Integrated with Self-attention Mechanism and Highway Network for Text Classification

Xuewei Li1* and Hongyun Ning2
Tianjin Key Laboratory of Intelligence Computing & New Software Technology, Tianjin University of Technology, Tianjin, 300384, China

*E-mail: 269114172@qq.com

Abstract. Text classification is one of the basic tasks of natural language processing. In recent years, deep learning has been widely used in text classification tasks. The representative one is the convolutional neural network. DPCNN is a deep convolutional neural network text classification model that can obtain long-distance text information, but it focuses on the extraction of global features and ignores the extraction of local features of the text. Some local feature information is very important and plays an important role in text classification tasks. Therefore, in this paper, a self-attention mechanism is introduced to extract local features of text based on the lack of extracting local features of DPCNN. In addition, although the deep convolutional neural network can extract deeper features, it is easy to cause the problem of gradient disappearance during training. Therefore, the highway network is introduced to prevent the problem of gradient disappearance caused by network training and improve the performance of the model. Experimental results show that the proposed model is better than a single DPCNN model, which further improves the accuracy of text classification tasks.

1. Introduction
Text classification is a fundamental problem in natural language processing (NLP). Conventional text classification methods use a shallow structure, and incorporate a large number of artificial features. Deep learning models have been widely used in this task in recent years. Such as recurrent neural network (RNN)[1] including long-term short-term memory (LSTM)[2] and gated recursive unit (GRU)[3] have been widely used in text processing due to their strong performance in processing text structures. Several variants were proposed[4][5] and achieved good results.

Convolutional neural networks (CNN) have achieved great success in the field of computer vision. In recent years, many scholars have applied convolutional neural networks to the field of NLP. Collobert and Weston[6][7] were the first to apply convolutional neural networks to the NLP field, and kim[8] was the first to apply convolutional neural networks to text classification tasks. Zhang[9] proposed character-level CNN models and reported competitive results. Kalchbrenner[10] proposed a CNN model that uses dynamic pooling. Johnson[11] proposed a semi-supervised convolutional neural network text classification model based on region embedding. The above models are relatively shallow convolutional neural network models. The convolution kernel uses a sliding window to extract text features, and can not capture long-distance text information. Therefore, Conneau[12] proposed a deep convolutional neural network text classification model VDCNN, which can learn long-distance
text information. Rie Johnson[13] proposed a similar deep convolutional neural network model DPCNN. Convolutional neural networks based on deep layers focus on the extraction of global features and cannot extract the local feature information of the document well. Therefore, this paper introduces a self-attention mechanism for the lack of local feature extraction by DPCNN. At the same time, this paper also introduces the highway network to solve the problem that the deep neural network easily causes the gradient to disappear, and further improves the accuracy of model text classification.

2. MODEL

2.1. Overview
This paper proposes a DPCNN text classification model incorporating self-attention mechanism. DPCNN focuses on the extraction of global features of text and cannot extract local features of text well. Based on the DPCNN model, we introduce a self-attention mechanism to extract local features of the text. At the same time, a highway network is introduced in each convolutional layer of DPCNN to prevent the problem of gradient disappearance, reduce the loss of information in each layer, and improve the performance of the entire model. The framework of our model is shown in Figure 1.

![Figure 1. Our proposed model](image)

2.2. DPCNN
DPCNN is illustrated in Figure 2. The first layer is a text embedding layer, followed by the stacking of convolutional blocks. Each convolutional block has two convolutional layers with shortcut connections. The convolutional layer uses a pre-activation method, pre-activation refers to activation being done before weighting, that is \( w\sigma(x) + b \). In addition, convolution blocks are connected with 1/2 pooling with stride = 2 for downsampling, and the final pooling layer performs the maximum pooling operation.

![Figure 2. DPCNN model](image)
2.3. Self-attention mechanism
The self-attention mechanism calculates the attention by learning the correlation between the words in
the text, which is mainly divided into three steps. First, calculate the similarity between the target
word and each word in the text to get the weight of each word. Second, use the softmax function to
normalize the obtained weights. Finally, the obtained weight coefficient and the text matrix are
weighted and summed to obtain the final attention value. The calculation process is as follows:

\[ X_{U A V}^{n*d} = \text{tanh} \left( \text{dot-Product} \right) \]

\[ A = \text{softmax}(X \ast U^T) \]  

\[ V_i = \sum_{j=1}^{n} a_{ij} \ast X_j \]

**Figure 3.** attention layer of Figure 1

As shown in Figure 3, the input text is a matrix of word vectors of \( n \times d \). First, \( X \) is fully
connected and then activated to obtain a new matrix \( U \). The calculation formula of \( U \) is as follows:

\[ U = \text{tanh}(WX + b) \]

Tanh is the activation function. The matrix of the activation matrix and the original word vector
matrix is used to perform the similarity calculation. Dot product similarity calculation method is used
here. Then the result of the dot product is normalized with softmax function to obtain the attention
matrix \( A, A \in R^{n \times n} \), each element in the matrix \( A \) is denoted as \( a_{ij} \), which represents the similarity
between each word in the text, that is, the weight coefficient between each word.

\[ A = \text{softmax}(X \ast U^T) \]

Each row element \( a_{ij} \) of the self-attention matrix \( A \) and each vector of \( X \) are weighted and summed
to obtain a text feature information vector \( V, v \in R^{n \times d} \) with attention weight, and \( V_i \) refers to the \( i \)
words with attention weight are calculated as follows:

\[ V_i = \sum_{j=1}^{n} a_{ij} \ast X_j \]

2.4. Highway network
We do not use the DPCNN pre-activation method, but use the traditional neural network post-
activation method. We introduce the highway network[14] as a non-linear activation layer in the
convolution layer. The refinement structure of the convolutional layer in Figure 1 is shown in Figure 4:

**Figure 4.** Conv layer with Highway network

The output of traditional neural network nonlinear conversion is generally \( y = g(wx + b) \), and
the highway network is based on the gate mechanism. transform gate \( t \) and carry gate \( c \) are introduced.

As shown in Figure 3, \( X \) is the input text vector. Convolution operation is performed on \( X \), and
the output passes through \( t \)-gate and \( g \)-gate, respectively. The \( t \)-gate is a sigmoid activation function and
the \( g \)-gate is a relu activation function. * in the figure represents the result of multiplying the two
outputs. Let \( c = (1-t) \). The final output formula in the figure 4 is as follows:
for the output, while the rest is non-linearly activated. Experiments show that this structure will help extract a lot of information.

2.5. Detailed description of the overall working principle of the model
As shown in Figure 1, the model extracts local and global features of the text in two paths.

1) In the first path, this paper introduces a self-attention mechanism, focusing on important word feature information, followed by a convolution layer and a pooling layer. The convolution layer's convolution kernel window size is 1, Scan the sentence and perform the convolution operation to extract the features of a single important word to obtain the feature vector C. The formula is as follows:

\[ z = t(wx + b) * g(wx + b) + c * x \]  

(4)

\[ V \] is a text matrix with attention weights, \( W \) is the weight of the convolution unit, \( b \) is the bias of the convolution unit, and \( \rho \) is a non-linear activation function. Where \( C = [C_1, C_2, \ldots, C_n] \).

Then perform the maximum pooling operation on the obtained feature vector in the pooling layer, and select the maximum feature value of the feature vector from the convolution layer, \( C^* = \max(C) \). The number of convolution kernels is 250, so the final text feature map is \( F = [C_1^*, C_2^*, \ldots, C_{250}^*] \).

2) In the second path, this paper uses the DPCNN model to extract the global features of the text. DPCNN convolutional layer uses pre-activation to prevent the problem of gradient disappearance. This paper does not use the original author's method, but uses the traditional neural network post-activation method. At the same time, a highway network is introduced in each convolutional layer of DPCNN to prevent the problem of gradient disappearance and enhance the performance of the model. The final text feature map is recorded as \( T = [T_1, T_2, \ldots, T_{250}] \).

3) This paper uses the concat function in the tensorflow framework to fuse the features obtained from the two paths, which is denoted as \( M = [C_1^*, C_2^*, \ldots, C_{250}^*, T_1, T_2, \ldots, T_{250}] \).

4) The last part of the model uses the softmax function to classify the obtained feature M. The formula is as follows:

\[ p(y/s) = \text{softmax}(WM + b) \]  

(5)

The label formula for text prediction is as follows:

\[ y^* = \arg \max_{y} p(y/s) \]  

(6)

The loss function used in this model is cross-entropy loss, and L2 regularization is introduced. the formula is defined as:

\[ J(\theta) = - \frac{1}{s} \sum_{i=1}^{s} t_i \log(p(y/s)) + \lambda \| \theta \|^2 \]  

(7)

where \( t \) is the probability of the true label of the text, \( p(y/s) \) is the probability of each class with softmax function, \( s \) is the number of target classifications, and \( \lambda \) is the L2 regularization hyperparameter, which represents the proportion of the model's complex loss in the total loss.

3. Experiment

3.1. Experimental environment
The experimental development language is Python 3.6, the development tool is pycharm, and the deep learning framework used is Tensorflow = 1.13.1. The Chinese word segmentation tool used is jieba = 0.39.

3.2. Experimental dataset
The dataset used in this experiment is a Chinese text classification dataset launched by the Natural Language Processing Laboratory of Tsinghua University. The THUCNews dataset contains 740,000 news texts with 14 candidate news categories. This paper uses 10 of these categories, taking 50,000 training set samples, 10,000 test set samples, and 5000 validation set samples.
3.3. Experimental parameters and training

The experimental model parameters are shown in Table 1. In this paper, for the THUCNews dataset, we use the unsupervised cbow model[15] pre-trained word vector to initialize the input word vector matrix.

Unsupervised pre-training of word vectors is an important ingredient in deep learning for NLP[8]. This experiment uses the Adam algorithm for model training. We use 50000 news texts for training and 5000 news texts for validation set. When the accuracy of the validation set cannot be improved for a long time, the training is stopped. At the same time, L2 regularization was used, which effectively prevented the problem of model training overfitting.

Table 1. Experimental parameters

| Parameters            | setting       |
|-----------------------|---------------|
| Kernel size           | 1, 3          |
| Number kernels        | 250           |
| Embedding size        | 100           |
| Batch size            | 64            |
| Seq_length            | 600           |
| Keep_prob             | 0.5           |
| Learning rate         | 0.001         |
| L2_reg_lambda         | 0.01          |

3.4. Results and Analysis

In order to verify the correctness and effectiveness of the experiment, this paper selects several classic deep neural network models in the field of text classification for comparative experiments. Comparison models include CNN, RNN, DPCNN.

Table 2. Accuracy for different neural models on THUCNews dataset

| Model                  | Accuracy | Loss  |
|------------------------|----------|-------|
| RNN                    | 94.22    | 0.21  |
| CNN                    | 96.73    | 0.12  |
| DPCNN                  | 97.05    | 0.1   |
| DPCNN+highway network  | 97.29    | 0.096 |
| DPCNN+self_attention   | 97.44    | 0.093 |
| Our model              | 97.53    | 0.086 |

As can be seen from Table 2, the accuracy of the models designed in this paper is better than the above-mentioned classic text classification models in THUCNews dataset. the test accuracy of the CNN is better than RNN, so we believe that CNN is more suitable for text classification tasks than RNN. DPCNN is a deeper convolutional neural network text classification model that can learn long-distance text information. The THUCNews dataset is a news text dataset with long text length. Compared to the shallow CNN model, DPCNN can extract deeper features of the text. We have also verified under THUCNews dataset that the DPCNN test accuracy is better than the shallow CNN model.

In order to illustrate the effectiveness of the model in this paper, the effects of self_attention mechanism and highway network on classification results are discussed. DPCNN, DPCNN + highway network, and DPCNN + self_attention are compared with our model. As shown in Table 2, by comparing DPCNN with DPCNN + highway network, it can be seen that the introduction of the highway network improves the accuracy rate by 0.24% under the THUCNews dataset. The highway network is based on a gate mechanism. Transform gate and carry gate are introduced, and the output is controlled by two gates. It not only prevent the problem of gradient disappearance caused by network training, but also reduces the loss of information in each layer of the network and adaptively optimizes
features, so the model classification results can be improved.

As shown in Table 2, by comparing DPCNN and DPCNN + attention, the DPCNN model with the self_attention mechanism increased by 0.39% on the THUCNews dataset. DPCNN can learn long-distance text information, ignoring the impact of local features of an article on text classification, and the self_attention mechanism can focus on important words in the article and extract local features of the text. Therefore, the DPCNN model with the self_attention mechanism can extract richer feature information and further improve the classification accuracy.

The model proposed in this paper, based on the DPCNN model, introduces a self-attention mechanism and a highway network. Compared with a single DPCNN model, the model proposed in this paper can extract richer text feature information. As can be seen from Table 2, the proposed model has the best classification effect under the THUCNews dataset, and the classification accuracy rate of the DPCNN model is improved by 0.48%. The experimental results show the effectiveness of the text classification task under this model.

4. Conclusion
This paper designs a text classification model combining DPCNN and self-attention mechanism, which extracts the global features and local features of the text in two ways. At the same time, based on the original DPCNN, a highway network is introduced to solve the problem of deep convolutional neural network training, which easily causes the gradient to disappear, control the flow of information between layers, and reduce the loss of text semantic feature information. Therefore, this model can extract rich text semantic features and improve the accuracy of text classification tasks. The model was evaluated on the news long text dataset and achieved good results, and experiments were designed to compare the effect of the self_attention mechanism and the highway network on the classification task. Experiments have also demonstrated that both have improved the accuracy of the model. Finally, the accuracy of the model test in this experiment reached 97.53%, which is better than other neural network models.

References
[1] Arevian G. Recurrent Neural Networks for Robust Real-World Text Classification[C]// IEEE/WIC/ACM International Conference on Web Intelligence (WT07). IEEE, 2008.
[2] Hochreiter S, Schmidhuber, Jü rgen. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735-1780.
[3] Cho K, Merrienboer B, Gulcehre C, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. 2014.
[4] Huang M, Qian Q, Zhu X. Encoding Syntactic Knowledge in Neural Networks for Sentiment Classification[J]. ACM Transactions on Information Systems, 2017, 35(3):1-27.
[5] Tai K S, Socher R, Manning C D. Improved semantic representations from tree-structured long short-term memory networks[J]. Computer ence, 2015, 5(1): 36.
[6] Ronan Collobert J W. J.Weston: “A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask[C]// 2008.
[7] Collobert R, Weston J, Bottou, Léon, et al. Natural Language Processing (Almost) from Scratch[J]. Journal of Machine Learning Research, 2011, 12(1):2493-2537.
[8] Kim Y. Convolutional Neural Networks for Sentence Classification[J]. Eprint Arxiv, 2014.
[9] Zhang X, Zhao J, Lecun Y. Character-level Convolutional Networks for Text Classification[J]. 2015.
[10] Kalchbrenner N, Grefenstette E, Blunsom P. A Convolutional Neural Network for Modelling Sentences[J]. Eprint Arxiv, 2014, 1.
[11] Johnson R, Zhang T. Semi-supervised Convolutional Neural Networks for Text Categorization via Region Embedding[J]. Adv Neural Inf Process Syst, 2015, 28:919-927.
[12] Conneau A, Schwenk H, Barrault, Loï c, et al. Very Deep Convolutional Networks for Text Classification[J]. 2016.
[13] Johnson, Rie & Zhang, Tong. (2017). Deep Pyramid Convolutional Neural Networks for Text Categorization. 562-570. 10.18653/v1/P17-1052.
[14] Srivastava R K, Greff K, Schmidhuber, Jü rgen. Training Very Deep Networks[J]. Computer ence, 2015.
[15] Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[J]. Computer ence, 2013.