A Novel Model Combining Transformer and Bi-LSTM for News Categorization

Yuanzhi Liu, Min He, Mengjia Shi, and Seunggil Jeon

***Abstract*— News categorization (NC), the aim of which is to identify distinct categories of news through analyzing the contents, has acquired substantial progress since deep learning was introduced into the natural language processing (NLP) field. As a state-of-art model, transformer's classification performance is not satisfied compared with recurrent neural network (RNN) and convolutional neural network (CNN) if it does not get pretrained. Based on the transformer model, this article proposes a novel framework that combines bidirectional long short-term memory (Bi-LSTM) network and transformer to solve this problem. In the suggested framework, the self-attention mechanism is substituted with Bi-LSTM to capture the semantic information from sentences. Meanwhile, an attention mechanism model is applied to focus on those important words and adjust their weights to solve the problem of long-distance information loss. With pooling network, the network complexity can be reduced and the main features can be highlighted by halving the dimension of the hidden state. Finally, after acquiring the hidden representation by the above structures, we utilize a contraction network to further capture the long-range associations from a text. Experiments on three large-scale corpora were performed to evaluate the suggested framework, and the results demonstrate that our model outperforms other models such as deep pyramid CNN (DPCNN), transformer.**

***Index Terms*— Attention mechanism, bidirectional long short-term memory (Bi-LSTM), natural language processing (NLP), news categorization (NC), transformer.**

I. Introduction

NEWS categorization (NC) is an application field for natural language processing (NLP). The task of NC is to extract the characteristics from raw texts and then predict their categories based on these features. However, with the exponential growth of information, it becomes harder to classify these news data by individual. Thus, NC technology has attracted increasing attention in recent years.

Some substantial advances have been made in deep learning. Word-embedding technology was proposed in [1] and [2] to promote the process of text analysis. In order to extract features automatically, Kim [3] applied the convolutional operation to capture the semantic representation from texts and enlarged the receptive field by increasing the max-pooling operation. Generally, convolutional neural network (CNN) can utilize different convolutional kernels to process blocks of sequences with different lengths. Furthermore, CNN was first adopted on large-scale multilabel text classification [4], which can extremely reduce the computational complexity and improve the robustness of expression. Some methods have attempted to deepen CNN to capture long-range associations from sentences [5], [6]. Unfortunately, a series of experiments [7] have been implemented to testify that merely adopting a deeper network cannot achieve better performance in most cases.

In some recent works, CNN and transformer are integrated to extract semantic features from sentences. For example, [8] designed a convolution network with an attention mechanism to enhance the ability to extract both local and global features from the text. Intuitively, this mechanism can capture important n-gram features from convolutional filter space. Then attention mechanism was utilized to generate the final representation by considering local and global information.

Different from traditional CNN, RNN [9] is a recurrent neural network that can mine temporal and semantic information effectively from data and share parameters in different parts of the model. Thus, RNN performs better in dealing with sequential data, but it fails to solve the problem of long-term dependencies due to the gradient disappearance. Long short term memory (LSTM) [10] and gated recurrent unit (GRU) [11] become substitutes to alleviate this problem through training the "gate" structure to hold or forget information. However, the sequence-dependent structure of RNN and its variants are hardly satisfied for high-efficiency parallel computing.

As a classifier, transformer [12] has been utilized widely in news text classification. Self-attention mechanism [13], [14], [15] can learn long-distance dependencies from the sentences while retaining local information, especially for long texts. However, the training time will greatly increase on large-scale corpora because the model's spatial complexity and temporal complexity are both $O(n^2)$, where n represents the sequence length. Furthermore, compared with LSTM - the natural sequential network, the transformer still lacks the ability of dimensional modeling even if the position encoding is adopted, therefore, the outputs at every position are similar to each other in a slightly deeper transformer model. Tang et al. [16] also observed that RNN has a better performance on capturing long-distance dependencies than Transformer while the distance between subject and predicate is greater than 13.

In this article, based on the transformer and bidirectional long short-term memory (Bi-LSTM), we propose a novel

Manuscript received 26 July 2022; revised 8 October 2022; accepted 6 November 2022. This work was supported by the Science and Technology Plan of Yunnan Province of China under Grant 2014AB016. *(Corresponding author: Min He.)*

Yuanzhi Liu, Min He, and Mengjia Shi are with the School of Information Science and Engineering, Yunnan University, Kunming 650091, China (e-mail: lyz1239949593@163.com; hemin@ynu.edu.cn; smjswc@163.com).

Seunggil Jeon is with Samsung Electronics, Yeongtong-gu, Suwon-si, Gyeonggi-do 16677, South Korea (e-mail: simon.sgjeon@gmail.com).

Digital Object Identifier 10.1109/TCSS.2022.3223621

network structure that utilizes the advantages of both transformer and RNN to overcome the above problems. Firstly, we apply Bi-LSTM to catch the long-distance dependencies from sentences, whose structure can reduce the model's complexity and process sequential information effectively. Meanwhile, we introduce an attention mechanism that can weigh the words and focus on some critical information in a sentence, which solves the problem of losing some important semantic information in a long text that Bi-LSTM might. Then, a contraction network is involved to enhance the prediction accuracy by capturing the long-range associations further from the hidden representation. The main contributions of this study are as follows.

1) We propose a novel structure that applies the Bi-LSTM network to replace the traditional self-attention network. This structure can set parameters to predict the time-series variables and also alleviate the problem of gradient disappearance that the RNN model has. In addition, it can also reduce the time complexity from $O(n^2)$ to $O(n)$, where n indicates the sequence length.
2) We attempt to assign different weights for words by incorporating the single-head attention mechanism to Bi-LSTM to capture long-distance dependencies better. In other words, the attention mechanism can focus on different locations to extract more semantic information, which gives more precise judgment without adding extra calculation and storage.
3) We design a contraction network to extract more global information by deepening the network without increasing the number of feature maps. Meanwhile, the residual structure [17] is utilized to guarantee performance when training some deeper networks. Concretely, we apply a convolution layer and sub-sampling operation to build a network, which can perceive more information while the number of internal parameters shrank.
4) We introduce a pooling network to compress the sequence length and sharply reduce the number of high-level parameters. Furthermore, we apply spatial linear mapping to strengthen the expression of the words.

II. Related Work

A. Attention Mechanism

Attention mechanisms [18] are mainly categorized into hard attention mechanism, soft attention mechanism, and self-attention mechanism, in which self-attention mechanism is widely utilized in text processing. The soft attention mechanism can decide the importance of each word by giving it a weight ranging from 0 to 1. However, it will cost an immense computational price when the input is long. The hard attention mechanism assigns a weight of either 0 or 1 for each input word to discard some irrelevant information. However, the backpropagation algorithm cannot be utilized to train the network since the hard attention mechanism adopts maximum or random sampling to select information. Self-attention mechanism [12] assigns different weights for each word by calculating the relevance between every two words, which helps to implement parallel computation and deal with long-distance dependency for the long text inputs.

B. Weak Supervision

Recently, weak supervision has attracted extensive concern because it can reduce the workload when dealing with vast amounts of data. Meng et al. [19] applied a pseudo-document generator to generate a pseudo-labeled corpus and then built a self-training module to refine the model. Mekala and Shang [20] attempted to design a weakly supervised model similar to unsupervised classification [21]. To deal with the polysemy problem, they cluster the context vectors by using the k-means method. Then the contextualized corpus generated by the contextualized representation learning is applied to set pseudo labels for unmarked texts. However, these unsupervised methods result in extensive calculation after iteration, which inevitably causes manpower consumption.

C. Graph Convolutional Network (GCN)

Recently, with the growing development of graph neural network [22], [23], [24] proposed a graph convolution model to classify texts. The authors take words and documents as nodes to construct a text graph for a whole corpus, and then these nodes are sent to a Softmax classifier to implement classification. However, due to the problem of ordered information that GCN ignores, text GCN is not better than other benchmark models when performing the emotional classification (e.g., on MR corpus). Furthermore, both the slipping window and vector dimension can influence the result. It is hard to obtain the global word co-occurrence information effectively with a smaller window size, while it will lead to an edge between two nodes that are not closely related when the window size is too large.

D. Variant of Transformer

The self-attention mechanism needs to calculate the dot product for all the words, which leads to quadratic growth of the computational complexity with the input length. Star-transformer [25] adjusts the fully connected structure to allow adjacent words of sequences to interact directly, while those nonadjacent words transmit information indirectly through a shared relay node. Therefore, the sequence length can be reduced to $2n$ while retaining the ability to capture local features. However, the fully connected attention mechanism requires an immense training corpus compared with our model, which is not suitable for dealing with some small-scale corpora.

III. Our Methods

For the traditional transformer model, the self-attention mechanism cannot effectively capture the local feature from the sentence vector, which inevitably ignores some critical information in a sentence. In addition, the holistic performance of the transformer is lower due to lacking the ability to language modeling. Therefore, we design a novel framework based on the transformer model and the structure is shown in Fig. 1.

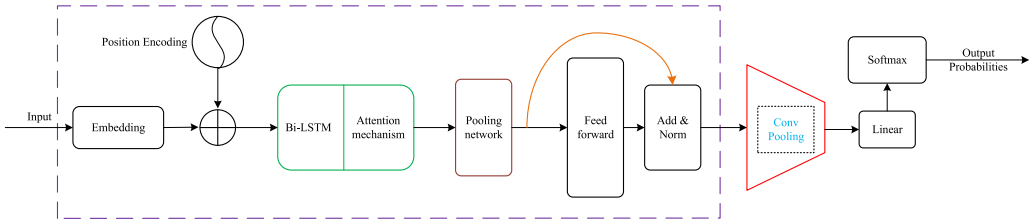


Fig. 1. Structure of our model. 1) Purple dashed curve is the encoder of the proposed model, $\oplus$ is the join operator. 2) Green and brown line indicates the improvement of transformer. 3) Red box represents contraction network that extracts features further from the hidden representation. 4) Output probabilities are the classification results predicted by a Softmax function.

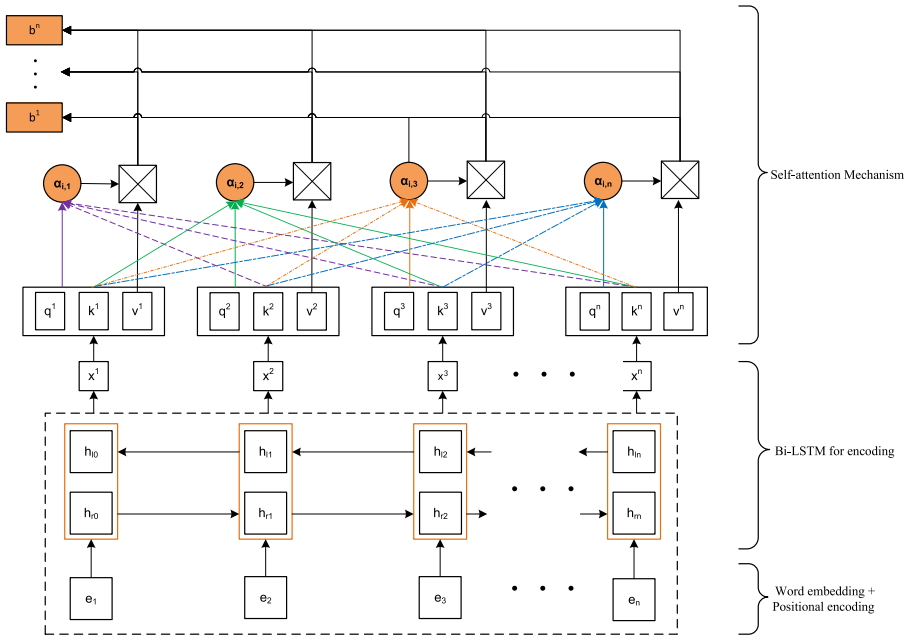


Fig. 2. Structure consists of Bi-LSTM and attention mechanism. $E_d = \{e_1, e_2, e_3, \ldots, e_n\}$ can be obtained by concatenating word embedding and positional encoding. The yellow line below represents the concatenation of the backward and forward hidden state. $\{q, k, \upsilon\}$ represents, respectively, {query, keys, value}.

Hypothesizing that $D = \{X^i\}_i^n$ indicates the sentence set in which each sentence is labeled with a certain category, and the input from D is sent to the proposed model to extract features. Firstly, after generating the sentence vector E_d by word embedding and positional encoding, Bi-LSTM is applied to encode E_d to acquire the hidden state Z. Then, the self-attention mechanism is adopted to capture the long-distance information from Z through those words with higher weights. Meanwhile, a pooling network is utilized to reduce the dimension of Z. Finally, the ultimate representation generated by the contraction network with a fully-connected layer is used to predict the sentence label. $p(k|\, D, \theta)$ represents the probability of the sentence that belongs to category k, where θ is the network parameter.

A. LSTM and Attention Mechanism

Different from the traditional transformer model, we choose Bi-LSTM as the feature extractor, which alleviates the long-term dependent problem that RNN has and can reduce the tremendous computational complexity caused by the multihead attention mechanism. However, Bi-LSTM still loses information if a sentence exceeds a certain length. In other words, Bi-LSTM cannot contribute to enhancing information retention.

Hence, we add a single-head self-attention mechanism into Bi-LSTM to capture as many features as possible. We calculate the correlation between every two words by constructing vectors (queries, keys, and values) to acquire the internal relationship of sentences. Thus, the attention mechanism can effectively avoid information loss, as illustrated in Fig. 2.

Given a sentence $X^i = \{y_1, y_2, \ldots, y_n\}$, each word y_i is converted to vector z_i by searching an embedding matrix $W^d \in \mathbb{R}^{B\times L}$, where B is the dimension of the embedding word, and L stands for vocabulary size. Assume $z_i = W^d * y_i$, where i represents the ith word in a sentence. Then z_i is sent to the positional encoding network to obtain its positional information in the input sequence. Suppose $E_d = e_i$ stands for the result of z_i after positional encoding. Following in [12], the formulas for constructing positional encoding are as follows:

$$\mathrm{PE}_{(\mathrm{pos},2i)} = \sin\left(\mathrm{pos}/10000^{2i/d_{\mathrm{im}}}\right) \tag{1}$$

$$\mathrm{PE}_{(\mathrm{pos},2i+1)} = \cos\left(\mathrm{pos}/10000^{2i/d_{\mathrm{im}}}\right) \tag{2}$$

where d_{im} indicates the dimension of network settings, i is the dimension of the vector.

We utilize a bidirectional LSTM containing a forward LSTM and a backward LSTM to generate the latent representation. $\{e_1, e_2, \ldots, e_n\}$ is sent to the forward LSTM to obtain the forward hidden representation $\{h_{l0}, h_{l1}, \ldots, h_{\mathrm{ln}}\}$ through encoding, and the backward LSTM generates a backward hidden representation $\{h_{r0}, h_{r1}, \ldots, h_{\mathrm{rn}}\}$ after importing vectors $\{e_n, e_{n-1}, \ldots, e_1\}$. Finally, the forward and backward hidden

representations are spliced to obtain Z: $\{h_{l0} + h_{r0}, h_{l1} + h_{r1}, \ldots, h_{\ln} + h_{\rm rn}\}$. Considering the advantages and disadvantages of the three attention mechanisms introduced in Section II, we adopt the single-head self-attention mechanism to acquire the critical information from the hidden representation Z. Concretely, we consider the weight of other words when calculating the attention score of one word through q and k, where q and k are the vectors produced according to hidden representation Z, the correlation between q_i and k_j is the attention score of other words to the word, where $j \in (1, n)$. According to the self-attention mechanism in [12], the calculative formulas are given by

$$\alpha_{i,j} = \frac{q^i k^j}{\sqrt{d}} \tag{3}$$

$$\alpha'_{i,j} = \frac{\exp(a_{i,j})}{\sum_{k=1}^{n} \exp(a_{i,k})} \tag{4}$$

where $i, j \in (1, n)$, d is the dimension of q and k to prevent dot product from being too large. Formula (4) is the numerical conversion of attention score by the Softmax function. The output of the self-attention mechanism is given by

$$b^i = \sum_{j=1}^{n} \alpha'_{i,j} v^j \tag{5}$$

where n indicates the number of words in a sentence, v is the produced vector according to Z. The weighted sum of different inputs can be considered by setting the value of $\alpha'_{i,j}$.

B. Pooling Network

Since the internal structure in the self-attention mechanism is all linear transformation, it inevitably weakens the expression of each word. To solve this problem, we map representation from low to high dimensions and then to low dimensions by applying the fully connected layer, the structure is illustrated in Fig. 3. Then, by using an rectified linear unit (ReLU) activation function, we strengthen the parts with large values and suppress the parts with small values in the word representation to enhance its expression.

In addition, the spatial information and the calculation are inevitably increasing when applying the attention mechanism to capture long-distance dependencies. Therefore, the pooling operation is utilized to shorten the dimensions of the hidden representation to reduce the number of parameters. In the article, we choose max-pooling to achieve better compression instead of average pooling, which is given by

$$X_C = \text{MaxPool}\left[\text{ReLU}\left(\text{Norm}\left(\left[X^i\right]_{\rm BA}\right)\right)\right] \tag{6}$$

where $[\ ]_{\rm BA}$ indicates the latent representation from Bi-LSTM and self-attention mechanism, X^i and Norm represent the ith sentence and normalization operation, respectively.

C. Add and Norm

Bi-LSTM, attention layer, and feed-forward layer have their own internal structure, respectively. Therefore, we apply a residual network to avoid gradient disappearance or gradient explosion. The formula is as follows:

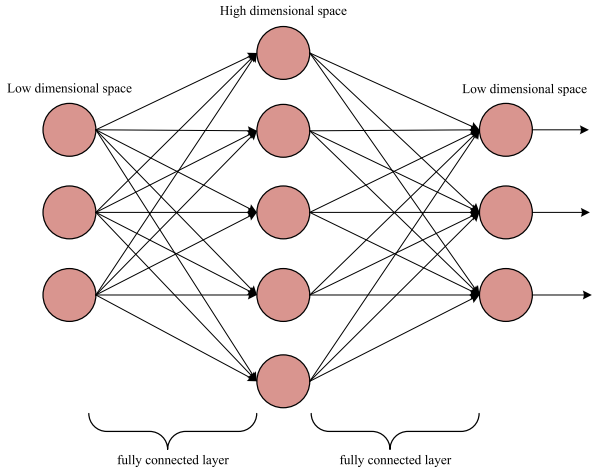


Fig. 3. First and third layers are low-dimensional space, higher dimensional space can be indicated by the second layer.

$$X_D = X_C + X_{\text{feed-forward}} \tag{7}$$

where $X_{\text{feed-forward}}$ indicates the output of X_C after the feed-forward network [12]. Then, we choose to standardize the representation again by normalization operation

$$X_{\text{hidden}} = \text{LayerNorm}(X_D). \tag{8}$$

D. Contraction Network

The previous works used to predict the category results directly after the encoder by connecting the linear layer. For the transformer model, it is not effective in capturing features without pretraining so it cannot acquire a satisfactory result, especially when the text length is long. Therefore, we deepen the network to capture more global information.

Concretely, we apply a 2-D convolution operation to enrich the expression of embedded words, and then the sequence length will be shortened to half of its original size through a max-pooling operation (stride = 2 and window size = 3). The convolution operation utilizes different sizes of filters over the word window to generate feature maps by

$$g^i = \sigma\left(u \cdot y^{i:i+t-1} + b\right) \tag{9}$$

where g^i and σ represent the feature of $y^{i:i+t-1}$ and activation function, respectively, $y^{i:i+t-1}$ is the word window in the sentence $\{y^{1:t}, y^{2:t+1}, \ldots, y^{n-t+1:n}\}$, u and b indicate the filter and bias, respectively.

Then, we implement a max-pooling operation over the feature maps to acquire the maximum value feature g' as the main feature of this filter, where $g' = \max\{g^1, g^2, \ldots, g^{n-t+1}\}$. Thus, the model could perceive twice as much information as the original sentence. As shown in Fig. 4, we reduplicatively utilize the module with convolution and max-pooling operation to perceive more information.

In addition, we discovered that increasing the number of feature maps will result in doubling the number of output

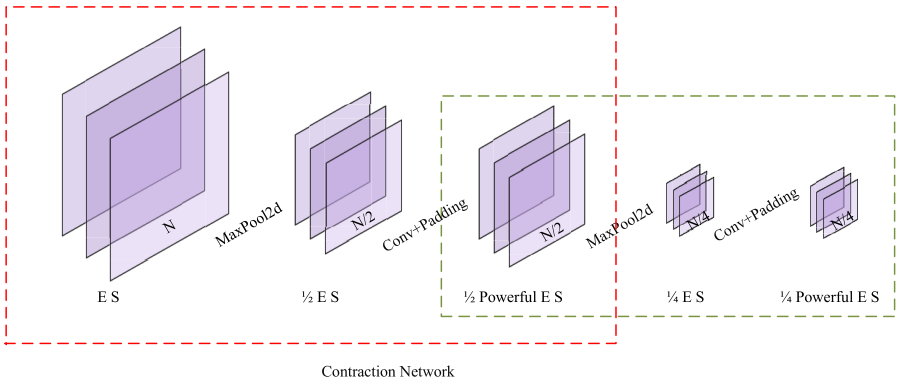


Fig. 4. ES represents the embedded sequence whose original length is N. The sequence length will be reduced to 1/2 of its original length after convolutional and pooling operations. Red and light green dashed curves indicate the primary circulation of the contraction network, which will cease circulation when the sequence length is less than 2.

channels. It was also demonstrated in [26] that adopting this method would enlarge the calculation but had no contribution to improving the classification accuracy. Thus, we fix the number of feature maps by adding the padding operation and then perform a max-pooling operation to halve the calculation time for each convolution layer. Furthermore, we also adopt the residual network to prevent gradient disappearance. The process can be depicted as

$$X_E = \text{MaxPool}\big[\text{Padding}(X_{\text{hidden}})\big] \tag{10}$$

$$H = \text{Conv}\big[\text{Relu}(\text{Padding}(X_E))\big] + X_E \tag{11}$$

where the plus sign stands for residual connection, and the final hidden representation H is used to predict the possible category results through a Softmax function

$$y_i'(j) = p\big(k \mid X^i\big) = \text{Softmax}\big(W^i H + b^i\big) \tag{12}$$

where d_i represents the ith sentence, W and b indicate weight and bias, respectively.

We choose cross-entropy loss as our loss function and then train parameters to minimize the loss function. The training loss is given by

$$L\big(y_i, y_i'\big) = -\sum_{j=1}^{m}\sum_{i=1}^{n} y_i(j)\log\big(y_i'(j)\big) \tag{13}$$

where $y_i(j)$ is the real label, $y_i'(j)$ denotes the predicted label, n represents the number of sentences, and m is the number of categories.

IV. Experience

To testify the efficiency of our model, we performed experiments on three corpora whose text length and number of categories are different from each other. Meanwhile, some comparative experiments are implemented on the same corpora.

A. Datasets

Three corpora—The Chinese THUCNews, Amazon, and AG-News are applied to evaluate our model. The Chinese THUCNews, a subset of the news text classification corpus provided by Tsinghua NLP Group, includes ten categories of "Finance," "Economics," "realestate," "stocks," "Education," "technology," "society," "politics," "sports," "games," and "entertainment." Amazon is a binary classification corpus about product reviews that are labeled with "positive" or "negative," which can be used to verify the model's generalization on other types of corpora. AG-News corpus contains four types of English news—"world," "sports," "business," and "tech." The vocabulary size is about 500 000 on Amazon and "UNK" is used to replace some unusual words. The data statistics are listed in Table I.

TABLE I
Statistics of Datasets

Corpus	THUCNews	AG-News	Amazon
Train	180000	110000	360000
Test	10000	7600	40000
Dev	10000	10000	20000
classes	10	4	2
Words	4762	150797	500000

B. Parameter

We made a number of readjustments to the parameters. The layer of the encoder is set to 2, and the word embedding matrix W^d is initiated with 300 dimensions. The learning rate is set to 0.0005, and 128 sentences are fed to the model to train at a time. The sentence length is set to 32, 45, and 100 on these three corpora, and the hidden dimension of the encoder is given by 150.

The dropout layer is appended to prevent overfitting and enhance its generalization. We have tested different values and discovered that 0.5 is a better setting while running on limited computing resources. The above parameters are the same for all models to ensure fairness. Due to limited memory, we set the batch size of sentences to 64 and 50 on the AG-News corpus and Amazon corpus for the Bert model. The best result can be acquired when training loss reaches convergence.

C. Evaluation Metrics

1) Confusion Matrix: We utilize a confusion matrix to summarize the number of correct predictions and incorrect

TABLE II
CONFUSION MATRIX

True value \ Predicted value	Positive	Negative
Positive	*TP*	*FN*
Negative	*FP*	*TN*

predictions. At the same time, we also detect which parts of the matrix will be confused when the model makes a prediction. We take two-classification as an example.

As shown in Table II, the first letter of TN, FP, FN, and TP represents whether predicted value is consistent with the actual value, where *T* and *F* represent the correct value and the wrong value, respectively. The second letter is the result of prediction, where *P* and *N* indicate positive samples and negative samples, respectively. Thus, the experimental model performs better if the values on the left diagonal are larger.

Although the confusion matrix can be used to judge the total accuracy rate, it generates misleading results in the case of imbalanced samples. For example, in a corpus that includes 90% of positive samples and 10% of negative samples, we can acquire a high accuracy rate of 90% by simply setting all the samples as positive. Therefore, we also use accuracy rate, precision, recall rate, and F1-score as evaluation metrics as follows.

a) Accuracy: We adopt test accuracy as our primary performance indicator to evaluate the classification accuracy, given by

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{14}$$

b) Precision: Precision is utilized to represent the rate of true positive samples to the identified positive samples, given by

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{15}$$

c) Recall: Recall indicates the proportion of true positive samples to the identified correct samples. Intuitively, a larger value means the model can correctly predict more positive samples, given by

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{16}$$

d) F1-Score: F1-score is the harmonic mean of precision and recall. The value of the F1-score ranges from 0 to 1, where 1 and 0 represent the best output and the worst output, respectively, given by

$$F1 - \text{Score} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}. \tag{17}$$

D. Results and Analysis

The evaluation results are listed in Table III and the analyses are as follows.

1) We noticed that CNN reaches a TA score of 0.9106 on THUCNews, and its recall, precision, and F1-score scores are around 0.91. However, the scores on AG-News and Amazon are 0.8937 and 0.9081, respectively, which is relatively low. The main reason is that the gradient descent algorithm makes the training results of CNN converge to a local minimum rather than the global minimum. Furthermore, the shallow CNN cannot capture long-distance dependencies, which can be demonstrated by the low score on Amazon.
2) RNN always outperforms the CNN model on AG-News and Amazon with TA scores of 0.8989 and 0.9207, respectively, which shows that the RNN model is more appropriate for capturing contextual information. Compared to RNN, the values of classification accuracy on all three corpora have been promoted by using LSTM, which testifies that LSTM can capture global information further through training the "gate" structure. However, LSTM just decreases information loss slightly. Although adding an attention mechanism can further solve this problem, LSTM + attn [27] achieves a TA score of 0.9245 on Amazon and the improvement is only about 0.1%. It's probably due to its weak parallel processing.
3) Because a deeper network can capture long-distance dependencies while CNN couldn't, therefore deep pyramid CNN (DPCNN) model [28] is utilized to perform experiments on these corpora. It is observed that the TA scores are only improved by 0.15% and 0.1% on THUCNews and Amazon, respectively, while the CNN model has a TA score of 0.8891 on AG-News. It indicates that the learning performance of deep CNN is degraded. Furthermore, a deep CNN network will increase the training time due to the complexity of the network. The pooling layer also loses some valuable pieces of information, which may cause the correlation between local and global information to be ignored by DPCNN.
4) As can be seen from Table III, transformer model cannot acquire satisfied results without pretraining. It obtains the worst score of 0.8988 and 0.8873 on THUCNews and Amazon, respectively. The main reason can be attributed to the fact that the self-attention mechanism focuses more on global information rather than local information, which proves the transformer model does not perform well on the task of text classification.
5) Bert [29] model achieves the best performance on THUCNews and Amazon corpora, with a TA score of 0.9470 and 0.9377, respectively. The high TA score of 0.9029 on AG-News also proves that the Bert model has universality for text classification. However, its training time is *n* times that of other models due to a large amount of computation. Furthermore, the Bert model usually requires an immense amount of tensor processing unit (TPU) resources when training data. Obviously, it has low price performance on some small corpora.
6) Our framework achieves the best TA score on AG-News with 0.9179, and the TA scores on THUCNews and Amazon are also higher than other models except the Bert model. It is not difficult to determine that the frame of combining LSTM and attention mechanism can focus

TABLE III

EVALUATION RESULTS OF OUR METHOD AND OTHERS ON THE TEST CORPORA, TA REPRESENTS TEST ACCURACY, ATTN INDICATES THE ATTENTION MECHANISM MODEL. (BOLDFACE INDICATES THE HIGHEST SCORE)

Data Set	THUCNews				AG-News				Amazon			
Indicator	TA	Precision	Recall	F1-Score	TA	Precision	Recall	F1-Score	TA	Precision	Recall	F1-Score
CNN	0.9106	0.9113	0.9106	0.9108	0.8937	0.8937	0.8937	0.8935	0.9081	0.9081	0.9081	0.9081
RNN	0.9085	0.9093	0.9085	0.9084	0.8989	0.8985	0.8989	0.8986	0.9207	0.9207	0.9207	0.9207
LSTM	0.9107	0.9109	0.9107	0.9105	0.9028	0.9032	0.9028	0.9028	0.9233	0.9237	0.9233	0.9232
LSTM+attn	0.9051	0.9068	0.9051	0.9054	0.8851	0.8875	0.8851	0.8852	0.9245	0.9246	0.9245	0.9245
DPCNN	0.9121	0.9129	0.9121	0.9122	0.8884	0.8891	0.8884	0.8881	0.9181	0.9181	0.9181	0.9180
Transformer	0.8988	0.8986	0.8988	0.8985	0.9005	0.9026	0.9005	0.9005	0.8873	0.8878	0.8873	0.8873
Bert	**0.9470**	**0.9473**	**0.9470**	**0.9471**	0.9029	0.9040	0.9029	0.9029	**0.9377**	**0.9377**	**0.9377**	**0.9377**
Ours	0.9151	0.9149	0.9151	0.9148	**0.9179**	**0.9181**	**0.9179**	**0.9178**	0.9297	0.9297	0.9296	0.9296

TABLE IV

ABLATION STUDY ON THE THUCNEWS CORPUS. (BOLDFACE INDICATES THE HIGHEST SCORE)

Variants	TA	Precision	Recall	F1-score
Transformer	0.8988	0.8986	0.8988	0.8985
+Bi-LSTM	0.9075	0.9072	0.9075	0.9069
+Bi-LSTM+ attention model	0.9079	0.9080	0.9079	0.9077
+Bi-LSTM+ attention model+ contraction network	0.9108	0.9117	0.9108	0.9109
+Bi-LSTM+ attention model+ contraction network+ pooling network	**0.9151**	**0.9149**	**0.9151**	**0.9148**

on both local and global information, which contributes to better-capturing features. And the convolution operation in the contraction network can further deepen the network to improve classification accuracy. Moreover, the pooling network can shorten the dimension of the hidden state, so our training time is much shorter than Bert and transformer. Finally, the values of accuracy rate are risen compared with the original transformer model on all the corpora, which demonstrates the effectiveness of the proposed framework.

E. Ablation Study

In this experiment, we apply the transformer model as a baseline to study the contributions of Bi-LSTM, attention mechanism, pooling network, and contraction network. In order to facilitate readers' observation, we implemented an ablation experiment with our framework on THUCNews corpus.

As shown in Table IV, a terrible TA score of 0.8988 was acquired by only applying transformer model. The TA score can be increased from 0.8988 to 0.9075 by replacing the self-attention mechanism with Bi-LSTM, which indicates that Bi-LSTM is literally suitable for sequence data due to its structure. In addition, the attention mechanism upgrades the performance by 0.04%. The results show that introducing attention mechanism into the model cannot improve the classification accuracy effectively. The reason is that the texts in THUCNews corpus are so short that Bi-LSTM wouldn't lose much information. Meanwhile, applying a contraction network could increase the TA score from 0.9079 to 0.9108, which illustrates deepening the network is effective. Besides, the proposed framework achieves a new state-of-the-art TA score of 0.9151 by adding a pooling network to the improved model. It shows that the pooling network can improve classification accuracy while reducing computational complexity. In conclusion, as demonstrated by the ablation experiment, all the improved structures contribute to promoting the performance of feature extraction.

V. CONCLUSION

In this article, we propose a transformer-based model to enhance the performance of NC. We apply Bi-LSTM and self-attention mechanism to replace the multihead attention mechanism to extract local and global features from the words that are assigned different weights. In addition, we adopt a pooling network to shorten the sequence length and then enlarge the receptive field to enhance the word representation, which also improves the classification accuracy. Then we use a contraction network to capture more global information. Experiments demonstrate that our framework has achieved better generalization performance on both Chinese and English news corpora than other models and possesses the advantage of not having a large number of calculations.

REFERENCES

[1] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. Int. Conf. Learn Represent.*, 2013, *arXiv:1301.3781*.

[2] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. 26th Int. Conf. Neural Inf. Process. Syst.*, vol. 2, 2013, pp. 3111–3119.

[3] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1746–1751.

[4] J. Liu, W.-C. Chang, Y. Wu, and Y. Yang, "Deep learning for extreme multi-label text classification," in *Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Aug. 2017, pp. 115–124.

[5] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[6] R. Johnson and T. Zhang, "Effective use of word order for text categorization with convolutional neural networks," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2015, pp. 103–112.

[7] H. T. Le, C. Cerisara and A. Denis, "Do convolutional networks need to be deep for text classification?" in *Proc. Workshops 32nd AAAI Conf. Artif. Intell.*, 2018. [Online]. Available: https://arxiv.org/abs/1707.04108
[8] P. Li et al., "ACT: An attentive convolutional transformer for efficient text classification," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, 2021, pp. 13261–13269.
[9] H. Zhang, L. Xiao, Y. Wang, and Y. Jin, "A generalized recurrent neural architecture for text classification with multi-task learning," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 2873–2879.
[10] R. Wang, Z. Li, J. Cao, T. Chen, and L. Wang, "Convolutional recurrent neural networks for text classification," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 2267–2273.
[11] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1422–1432.
[12] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5998–6008.
[13] E. Voita, D. Talbot, F. Moiseev, R. Sennrich, and I. Titov, "Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 5797–5808.
[14] P. Michel, O. Levy, and G. Neubig, "Are sixteen heads really better than one?" in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 14037–14047.
[15] K. Clark, U. Khandelwal, O. Levy, and C. D. Manning, "What does BERT look at? An analysis of BERT's attention," in *Proc. ACL Workshop BlackboxNLP, Analyzing Interpreting Neural Netw. (NLP)*, 2019, pp. 276–286.
[16] G. Tang, M. Müller, A. Rios, and R. Sennrich, "Why self-attention? A targeted evaluation of neural machine translation architectures," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 4263–4272.
[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[18] A. D. S. Correia and E. L. Colombini, "Attention, please! A survey of neural attention models in deep learning," *Artif. Intell. Rev.*, vol. 55, no. 3, pp. 1–88, Mar. 2022.
[19] Y. Meng, J. Shen, C. Zhang, and J. Han, "Weakly-supervised neural text classification," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2018, pp. 983–992.
[20] D. Mekala and J. Shang, "Contextualized weak supervision for text classification," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 323–333.
[21] Z. Haj-Yahia, A. Sieg, and L. A. Deleris, "Towards unsupervised text classification leveraging experts and word embeddings," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 371–379.
[22] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 3844–3852.
[23] H. Cai, V. W. Zheng, and K. C. Chang, "A comprehensive survey of graph embedding: Problems, techniques, and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 9, pp. 1616–1637, Feb. 2018.
[24] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 7370–7377.
[25] Q. Guo, X. Qiu, P. Liu, Y. Shao, X. Xue, and Z. Zhang, "Star-transformer," in *Proc. Conf. North*, vol. 1, 2019, pp. 1315–1325.
[26] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for natural language processing," *Künstliche Intelligenz*, vol. 26, Jun. 2016. [Online]. Available: https://arxiv.org/abs/1606.01781
[27] P. Zhou et al., "Attention-based bidirectional long short-term memory networks for relation classification," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 207–212.
[28] R. Johnson and T. Zhang, "Deep pyramid convolutional neural networks for text categorization," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 562–570.
[29] J. D. M.-W. C. Kenton and L. K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.

Yuanzhi Liu received the B.S. degree from Linyi University, Linyi, China, in 2018. He is currently pursuing the M.S. degree with the School of Information Science and Engineering, Yunnan University, Kunming, China.

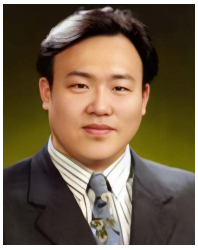
His main research interests include natural language processing and knowledge graph.

Min He received the B.S. and M.S. degrees in computer application technology from Liaoning Technical University, Fuxin, China, in 1998 and 2001, respectively, and the Ph.D. degree in computer science and technology from the University of Electronic Science and Technology of China, Chengdu, China, in 2006.

Now, she is an Associate Professor and the M.S. Supervisor with Yunnan University, Kunming, China. Her main research interests include intelligent information process, social network analysis, and embedded system application.

Mengjia Shi received the B.S. degree from Qingdao Technological University, Qingdao, China, in 2021. She is currently pursuing the M.S. degree with the School of Information Science and Engineering, Yunnan University, Kunming, China.

Her main research interests include natural language processing and knowledge graph.



Seunggil Jeon received the B.S. and M.S. degrees from Konkuk University, Seoul, South Korea, in 2001 and 2003, respectively, and the Ph.D. degree from Hanyang University, Seoul, in 2008.

He is currently a Principal Engineer at Samsung Electronics, Suwon, South Korea.