IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 25, 2020, accepted November 17, 2020, date of publication November 20, 2020,
date of current version December 7, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3039470

Transformer Based Multi-Grained Attention Network for Aspect-Based Sentiment Analysis

JIAHUI SUN[1], PING HAN[2], ZHENG CHENG[1], ENMING WU[1], AND WENQING WANG[2]
[1]Basic Experiment Center, Civil Aviation University of China, Tianjin 300300, China
[2]College of Electronic Information and Automation, Civil Aviation University of China, Tianjin 300300, China

Corresponding author: Ping Han (hanpingcauc@163.com)

This work was supported by the Fundamental Research Funds for the Central Universities through the Special Project of the Civil Aviation University of China under Grant No. 3122019114.

ABSTRACT Aspect-based sentiment analysis aims to predict sentiment polarity for every aspect in a sentence review. Most existing approaches are based on the sequence models, which may superimpose the emotional semantics of different tendencies and lack syntactic structure information. And most models adopt coarse-grained attention mechanism which still face the issues of weakness interaction between aspect and context. In this paper, we propose a transformer based multi-grained attention network (T-MGAN), which utilizes the Transformer module to learn the word-level representations of aspects and context respectively, and further utilizes the Tree Transformer module to obtain the phrase-level representations of contexts. It is capable of extracting the syntactic structure features and syntax information of aspect and context. In addition, we adopt dual-pooling method and multi-grained attention network to extract high quality aspect-context interactive representations. We evaluate the proposed model on three datasets and prove the effectiveness of the proposed model.

INDEX TERMS Aspect-based sentiment analysis, transformer, tree transformer, attention mechanism, nature language processing.

I. INTRODUCTION

Sentiment analysis is one of the important tasks in natural language processing (NLP) that use computer-aided algorithm to obtain subjective feelings such as opinions and evaluations held by people on products, services, events and other objects [1]. In addition to giving the overall evaluation, people also usually make evaluations from multiple perspectives of entities, resulting in multiple sentiment polarities in a single review. For instance, in the review of restaurant, "*The food was definitely good, but the price is too high.*", sentiments expressed for the aspect "*food*" and "*price*" are positive and negative respectively. If the traditional sentiment analysis approaches are still adopted to classify the whole sentences' sentiment polarity, the results may be biased. In view of this, researchers proposed aspect-based sentiment analysis (ABSA) approach, which is aimed at investigating sentiment polarity with each specific aspect or target in the given review sentences [2]–[6]. Therefore, ABSA has become one of the key subtasks of sentiment analysis gradually.

The associate editor coordinating the review of this manuscript and approving it for publication was Ali Shariq Imran.

In recent years, deep learning-based algorithms, especially neural network models, have been making new progress in NLP tasks, and there are several methods have good performances in ABSA tasks [7]–[10]. The model based on the convolutional neural network (CNN) [11]–[13] can obtain the dependencies between words and semantic features of the sentences by using a window-fixed filter. However, the important structural information of sentences cannot be extracted. The model based on Recurrent neural network (RNN) and its derived models, such as long short-term memory (LSTM) [14] and gated recurrent units (GRU) [15], treat sentences as word sequences and can obtain the effective syntactic level features. Therefore, these sequential models have achieving competitive results in ABSA task [16], [17]. However, the mechanism of long short-term memory adopted by such models will lead to the superposition of emotional semantics. When there are multiple specific aspects with inconsistent sentiment polarity in a sentence, the resolution of the model will be affected, and the dependency between words in the sentence will be weakened with the increase of distance. The attention mechanism can effectively focus on the important information, so the combination of neural

networks such as CNN or RNN with the attention mechanism enables the model to focus on the important features, which determining the sentiment polarity of corresponding specific aspects in the context [18], [36]–[38]. Although this kind of model can pay more attention to some important features in training, it usually uses a single attention mode which makes it impossible for the model to deeply extract the interaction between specific aspects and context. In addition, there are two problems with these models: First, when a particular aspect is not a single word but a phrase, they usually take the average vector of several words as the representation of a particular aspect [36]. Although this method is relatively simple, it cannot fully reflect the characteristics of each word in the phrase, and resulting in the loss of useful information. Second, when further obtaining the interaction characteristics between a specific aspect and context, they usually pool the specific aspect/context feature matrix and learns the attention weight of each word in the context/specific aspect respectively after average pooling [18], [19]. However, the use of single pooling will result in the loss of some useful information.

To tackle above issues, this paper proposes an aspect-based sentiment analysis model: Transformer based multi-grained attention network (T-MGAN). The proposed model utilizes the Transformer module [20] to learn the word-level representations of aspects and contexts respectively, and further utilizes the Tree Transformer module [21] to obtain the phrase-level representations of contexts. The Transformer is a new sequence transduction model without recursive and convolutional structures [20]. Its core unit is the self-attention mechanism that performs feature calculation for every two words in the sentence. The ability of the Transformer module to acquire features will not decrease as the distance between two words increase, and this module calculates a more comprehensive attention representation after multiple linear transformations, which can avoid the problem that CNN or RNN combined with a single attention mechanism leads to insufficiently comprehensive representations. In addition, human language has a hierarchical structure. Although the Transformer module has advantages in acquiring word-level features, it lacks phrase-level features that reflect the language hierarchy and syntactic information. When there are transitions or negative words in a sentence, if phrase-level features are not considered, it may lead to wrong judgments of sentiment polarity in corresponding aspects. For example, in the sentence "*The server served us without a smile*", if the phrase "*without a smile*" is not considered but only "*smile*" is considered, the judgment result of the sentiment polarity of the sentence will be changed. Wang *et al.* [21] proposed a Tree Transformer model for syntactic analysis tasks, which is excellent in determining the grammatical structure of sentences and the dependency relationship between words. Inspired by this model, we combine the Tree Transformer model with the specific situation of aspect-based sentiment analysis task to learn the phrase-level representations of context, and further improve the problem of insufficient features caused by the previous model using only word-level representations. In general, to the best of our knowledge, we are the first one to apply the Tree Transformer module to aspect-based sentiment analysis. In addition, in order to avoid the information loss caused by using the simple average aspect vector as the representation to learn the attention weights on the context words, we calculate the attention score for each word in the aspect, so as to obtain the more effective representation for sentiment analysis. Furthermore, in order to avoid the loss of some useful information caused by single-pooling, this paper adopts dual-pooling method.

To sum up, the main contributions of this paper are as follows:

- This paper proposes a novel aspect-based sentiment analysis model T-MGAN, which based on the Transformer module and the tree structure based Tree Transformer module. It can obtain both the word-level and phrase-level representations that contain syntactic structure and grammatical knowledge from different perspectives. The proposed model can solve the problem of insufficient feature extraction of existing models to a certain extent, and can obtain more effective information in a complex context than sequence models.
- This paper adopts the dual-pooling method to select the important features of aspect and context, which can reduce the loss of learned features. And then we utilize the attention mechanism multiple times to characterize the word-level interactions and phrase-level interactions between aspect and context words.
- In this paper, we evaluate this proposed model on three datasets that commonly used in the field of aspect-based sentiment analysis, and compare the experimental results with other models. The results show that our model is effective.

II. RELATED WORK

ABSA is an important branch of sentiment analysis task [22]. Its goal is to identify the sentiment polarity of corresponding aspects in a sentence. Compared with ordinary sentence-level or document-level sentiment analysis task, it requires a deeper level of fine-grained sentiment information [23], [24]. ABSA can be decomposed into two steps: the first step is aspects extraction with supervised [25] or unsupervised [26] methods, and then classify the extracted aspects into sentiment polarity. This paper mainly studies the classification methods of sentiment polarity given a specific aspect and does not study the extraction methods of aspects.

In the early research, the methods of ABSA are mainly based on rules [27] and traditional machine learning methods. As shown in literature [28], support vector machine (SVM) are trained by well-designed handcrafted features. These S methods can make good use of text information by constructing features, but they need a lot of preprocessing and complex feature engineering for input text, as well as external knowledge such as dependency parsing. The effect of the model largely depends on the quality of artificial feature design and the adequacy of prior knowledge, which requires more human source. With the successful application of distributed

representation learning method [29] in NLP tasks such as machine translation and automatic question answering, as well as the continuous improvement of multi-layer network architecture fitting and learning ability, researchers have transferred the traditional method relying on artificial feature engineering to deep learning method.

Xue and Li [30] proposed a gated convolutional neural network based model which utilize Gated Tanh-ReLU unit to selectively output sentiment polarity according to a given specific aspect, and achieve good results in training speed and classification accuracy. However, the filter of CNN in this model is limited by the filter window size, and the ability to obtain the feature between words with a long distance is weak, which makes the classification result unable to be further improved. Dong *et al.* [8] are the first one to employ the recursive neural network for ABSA on Twitter dataset, and use the syntactic structure information auxiliary model to improve the accuracy of sentiment analysis. However, this model has strong dependence on syntax, and the classification effect is greatly affected when the sentence syntax expression is not standardized. RNNs, such as LSTM and GRU have advantages in sequence modeling and are widely used in ABSA. Tang *et al.* [31] employed the forward and backward LSTM to model the context of the front and back parts of a specific aspect separately, and then combined the two parts representations as the overall characteristics for prediction. Ruder *et al.* [32] utilized a two-way LSTM network to effectively extract the features between words in an input sentence at different levels, but this method is only applicable to the case that the sentence contains only one specific aspect word. Due to the characteristics of RNN, it pays more attention to the recent input, but cannot capture the potential relationship between the relatively distant sentiment words or phrases and specific words in complex sentences. Therefore, researchers have introduced attention mechanism to solve this problem. Wang *et al.* [36] proposed an LSTM model combined with attention mechanism, which inputs the specific aspects of vectorization into the LSTM network, so that the model can focus on the feature information of specific aspects in the training process, so as to improve the result of sentiment classification. Chen *et al.* [33] proposed a RAM model which combines GRU network and attention mechanism. Trough nonlinear combination of features captured by different attention layers, the expression of sentiment polarity on specific aspects is obtained. The RNN based model has achieved a lot of good research results in ABSA. However, the resolution of the model will be affected by the superposition of sentimental semantic features of different polarity.

The Transformer module only composed of self-attention mechanism has achieved better results than CNN and RNN in many machine translation tasks. Recently, researchers have gradually applied this kind of model to ABSA. Zeng *et al.* [35] proposed the LCF model, which utilized the multi-head self-attention mechanism to capture local context features and global context features respectively, and added CDM layer to obtain the location information. When the Glove vector is used as input, the result is better than other models under the same conditions. In this paper, the proposed model not only employs Transformer module to extract word-level features for aspect and context, but also introduces the Tree transformer module to ABSA for the first time to obtain phrase-level features for context. In addition, in order to avoid the information loss caused by using the simple averaged aspect vector as the representation to learn the attention weights on the context words, we leverage both the *Aspect2Context* and *Context2Aspect* attentions to compose the Inter-Feature Multi-Attention Layer, which can obtain more comprehensive interaction features between aspect and context. Furthermore, in order to avoid the loss of some useful information caused by single-pooling, we adopt dual-pooling operation.

TABLE 1. Representation of different aspects of the same sentence.

Sentence	Aspect	Label
The food was definitely good, but the price is too high.	food	1
The food was definitely good, but the price is too high.	price	-1

III. THE PROPOSED MODEL

In this section, we describe the structure of the proposed model Transformer based multi-grained attention network (T-MGAN) as shown in Figure 1. It consists of the Input Word Embedding Layer, the Intra-Feature Extraction Layer, the Inter-Feature Multi-Attention Layer and the Output Layer.

A. TASK DEFINATION

Input of ABSA tasks includes two parts: the sentence to be determined and its corresponding specific aspects. And the model classifies the sentiment polarity of aspect term according to the characteristic information implied in the word sequence of the sentence. Suppose we are given a context sentence $S = \{x_1, \ldots, x_N\}$ and it corresponding one or more specific aspects $A = \{a_1, \ldots, a_k\}$, each of aspect may be a single word or a word phrase concludes $M \in [1, N-1]$ words $a_i = \{x_{i_1}, \ldots, x_{i_M}\}$, and A is a subset of S. The specific task of this paper is to infer the sentiment polarity (positive, negative and neutral) of specific aspects according to the input sentence S. For example, the sentiment polarity of "*food*" and "*price*" in the sentence "*The food was definitively good, but the price is too high.*". In order to better distinguish the sentiment polarity of different aspects in the same sentence, the task is expressed separately according to the number of specific aspect target words in the same sentence, as shown in Table 1.

B. INPUT WORD EMBEDDING LAYER

The word embedding layer maps each word into a low-dimensional real-value vector $v_i \in \mathrm{R}^{d_e}$, where d_e is

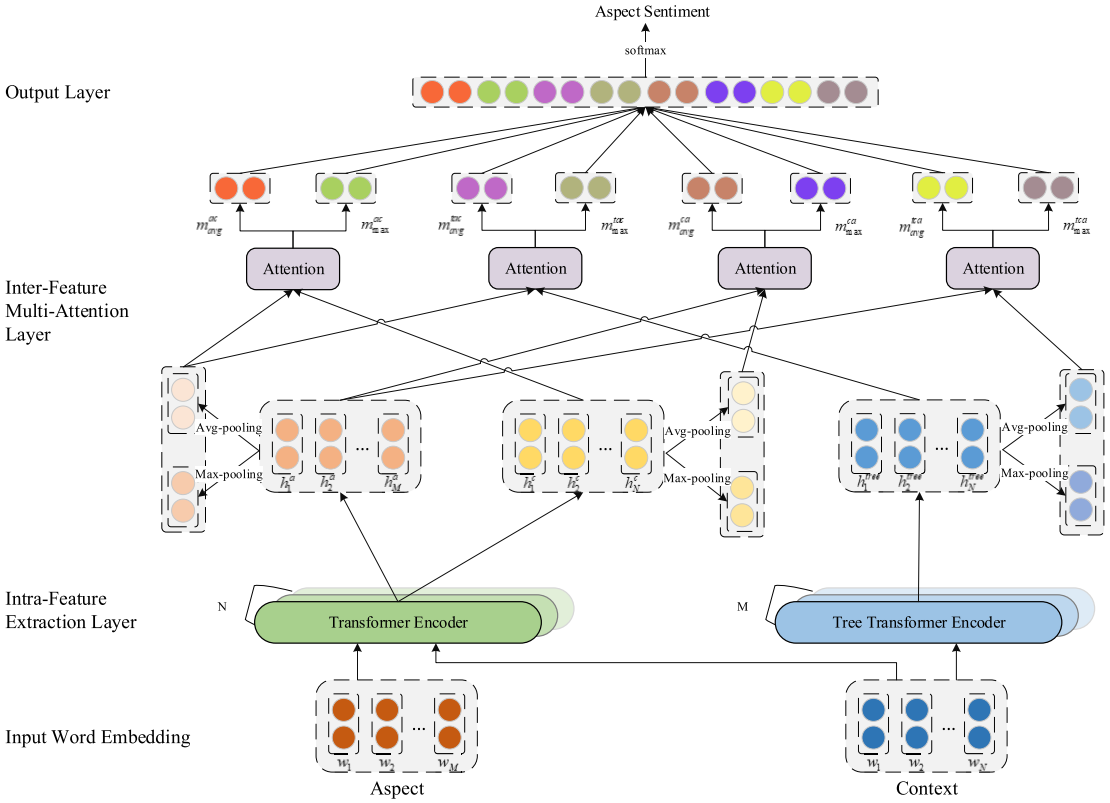

FIGURE 1. The architecture of the proposed T-MGAN model.

the dimension of vector. In T-MGAN design, Glove [39] word embeddings are alternatives for the embedding layer. We obtain the specific word vector through lookup embedding matrix $D \in \mathrm{R}^{|V|\times d_e}$ generated by Glove, where V is the size of the vocabulary. And then obtain the word vector matrix of context sentence sequence $\{w_1, \ldots, w_N\} \in \mathrm{R}^{N\times d_e}$ and the word vector matrix of aspect sequence $\{w_1, \ldots, w_M\} \in \mathrm{R}^{M\times d_e}$.

C. INTRA-FEATURE EXTRACTION LAYER

In order to obtain the word-level and phrase-level hidden representations for aspects and context, Transformer Encoder and Tree Transformer Encoder modules are adopted in this layer according to the characteristics of ABSA.

1) TRANSFORMER ENCODER

Transformer is a sequence-to-sequence transduction model, which is implemented by the encoder-decoder framework. The specific working method is to convert the input sequence into a fixed-length vector code through the encoder part, and then utilize the decoder part to convert the previously generated fixed vector converted into output sequence. However, the purpose of this paper is to vector encoding the original input context and aspect to obtain high-level features, there is no need to convert the encoded vector into sequence output, so only the encoder part of Transformer is used. As shown in Figure 2 (a), this part is composed of N identical

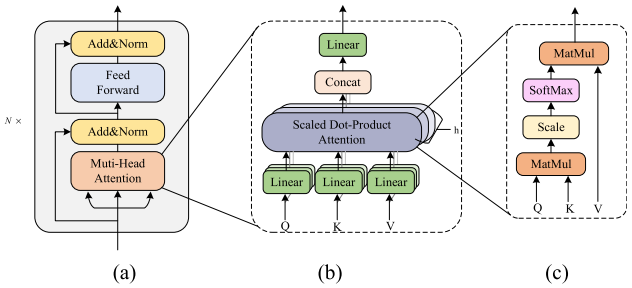

FIGURE 2. The architecture of transformer encoder.

layers, and each layer is composed of two sub-layers, which are multi-head attention mechanism and fully connected feed forward network. In addition, the residual connection and normalization operations are added after the two sub-layers. Among them, the multi-head attention is obtained by stacking multiple scaled dot-product attention, as shown in Figure 2 (b) and (c).

The input of Transformer Encoder is a matrix $X \in R^{M\times d_e}$ containing M words obtained by Input Word Embedding Layer. Three linear transformation matrices $W^Q \in R^{d_e\times \frac{d_h}{h}}$, $W^K \in R^{d_e\times \frac{d_h}{h}}$, $W^V \in R^{d_e\times \frac{d_h}{h}}$ are randomly initialized and multiplied with input matrix to obtain query matrix $Q = (q_1, \ldots, q_M)$, key matrix $K = (k_1, \ldots, k_M)$ and value matrix $V = (v_1, \ldots, v_M)$, where $q_i, k_i, v_i \in R^{\frac{d_h}{h}}$, d_h is

hidden dimension. The key step in Transformer Encoder is scaled dot-product attention. Firstly, the similarity between each vector q_i in the query matrix and each vector in the key matrix is calculated, and then the similarity vector is normalized to get the weight. Finally, the weight vector is multiplied by the value of all words in the sentence to get the output of scaled dot-product attention:

$$\begin{aligned} head_j &= Attention(q_{ji}, K_j, V_j) \\ &= soft\max(\frac{q_{ji}K_j^T}{d}) \cdot V_j \end{aligned} \tag{1}$$

where the scaling factor d is usually set to the square of the vector dimension $\sqrt{d_k}$ in the K matrix.

After h-order linear transformation of query, key and value matrix with different parameters, much more abundant features can be obtained by learning different groups for many times. And then the output of the multi-head attention mechanism is as follows

$$MHSA = (head_1 \oplus head_2 \oplus ... \oplus head_h) \cdot W_O \tag{2}$$

where "$\oplus$" denotes vector concatenation, $W_O \in R^{d_h \times d_h}$. Word embedding of aspect and context are taken as the input of Transformer Encoder respectively, and then calculate the hidden aspect representation $MHSA^a = [h_1^a, h_2^a, \ldots, h_M^a], h_i^a \in R^{d_h}$, as well as the word-level hidden context representation $MHSA^c = [h_1^c, h_2^c, \ldots, h_N^c], h_i^c \in R^{d_h}$.

2) TREE TRANSFORMER ENCODER

Context usually contains more words, complex grammatical components and syntactic structures. Therefore, this paper utilizes Tree Transformer module to further obtain the phrase-level features of context. This module can capture the phrase syntax information and the dependency between words in the context statement only by recursive traversal. The structure of the Tree Transformer is shown in Figure 3.

Based on the multi-head self-attention structure, the model adds the constant attention module to calculate whether words within a certain span can form a phrase. If the correlation probability calculated between two words is large, it can be considered that the two words constitute a phrase. The Tree Transformer module has a multi-layer structure, as shown in Figure 4, some sub phrases will be combined at each level, and several shorter phrase components will be gradually added from the lower level to the higher level. The key part of Tree Transformer is to calculate the constant priors C, and then get the component attention probability matrix E as follows:

$$E = C \odot a_i(Q, K) = C \odot soft\max(\frac{QK^T}{d}) \tag{3}$$

where "$\odot$"is element-wise multiplication, $C \in R^{N \times N}$, N is the number of words in the input sentence, E_{ij} is the probability of position i attends to j.

The constant priors C of each layer is different, but all the heads of multi-head self-attention in the same layer share the same one. It is obtained by predicting the probability that two

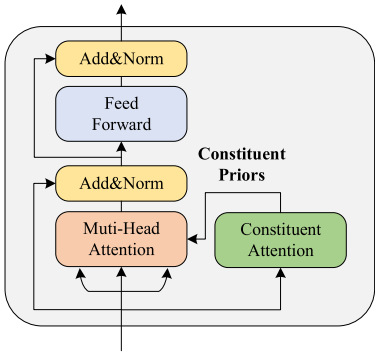

FIGURE 3. The architecture of Tree Transformer Encoder.

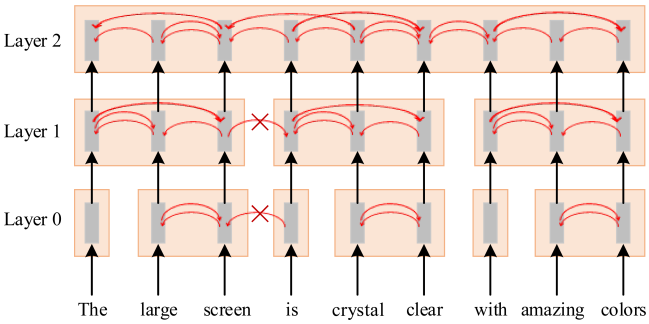

FIGURE 4. Schematic diagram of the multi-layer structure of tree transformer encoder.

adjacent parts belong to the same phrase. For the layer of Tree Transformer module, a sequence $a = \{a_1^l, \ldots, a_i^l, \ldots, a_N^l\}$ is defined, where a_i^l is the probability that w_i and w_j belong to the same component. The matrix C^l is calculated from all $a_{i \le k < j}^l$ between w_i and w_j according to formula 4:

$$C_{ij}^l = e^{\sum_{k=i}^{j-1} \log(a_k^l)} \tag{4}$$

If the two parts of a sentence belong to the same phrase at the lower level, they will have a greater probability of belonging to a longer phrase at the higher level. Therefore, the level of the sentence is not only related to the probability of the current layer, but also to the probability of the upper layer. The specific calculation method is as the following formulas:

$$a_k^l = a_k^{l-1} + (1 - a_k^{l-1}) \times \widehat{a}_k^l \tag{5}$$

$$\widehat{a}_k^l = \sqrt{p_{k,k+1} \times p_{k+1,k}} \tag{6}$$

$$p_{k,k+1}, p_{k,k-1} = soft\max(s_{k,k+1}, s_{k,k-1}) \tag{7}$$

$$s_{k,k+1} = \frac{q_k \cdot k_{k+1}}{h \times d_k/2} \tag{8}$$

In this paper, the component attention probability matrix E and the output of scaled dot-product attention are calculated as follows:

$$\begin{aligned} head_j^{tree} &= Tree\_Attention(q_{ji}, K_j, V_j) \\ &= E \cdot soft\max(\frac{q_{ji}K_j^T}{d}) \cdot V_j \end{aligned} \tag{9}$$

$$Tree\_MHSA^c = (head_1^{tree} \oplus head_2^{tree} \oplus ... \oplus head_h^{tree}) \cdot W_o^{tree} \tag{10}$$

Finally, the phrase-level implicit features of context are obtained $Tree\_MHSA^c = [h_1^{tree}, h_2^{tree}, \ldots, h_N^{tree}]$, $h_i^{tree} \in R^{d_h}$.

D. INTER-FEATURE MULTI-ATTENTION LAYER

The features of aspects as well as the word-level and phrase-level features of context have obtained through the Intra-Feature Extraction Layer. In this layer, the effective interaction features between aspects and context will be further obtained through multi-attention mechanism. When calculating the attention mechanism of pooled features *Pool* relative to the feature matrix$H = [h_1, h_2, \ldots, h_n]$, in the first, the multiplicative model of attention mechanism is used to calculate the similarity between the feature matrix H and the pooled feature *Pool*, and then obtain a representation vector representing the similarity of features between corresponding positions. The calculation method is as following formula:

$$f(Pool, h_i) = \tanh(Pool \times W \times h_i + b) \tag{11}$$

Here tanh is an activation function, $W \in R^{d_h \times d_h}$ is the attention weight parameter matrix, $b \in R^{d_h}$ is the bias vector.

The attention weight vector $\alpha_i(Pool, h_i)$ is obtained by normalizing the vector $f(Pool, h_i)$ and the elements in $f(Pool, h_i)$ represent the correlation between the corresponding position features in H and *Pool*.

$$\alpha_i(Pool, h_i) = \frac{\exp(f(Pool, h_i))}{\sum_{j=1}^{n} \exp(f(Pool, h_j))} \tag{12}$$

The weight matrix and the original feature matrix are weighted sum to obtain the associated feature representation:

$$m = \sum_{i=1}^{n} \alpha_i \cdot H_i \tag{13}$$

Due to the loss of some important information caused by single pooling, this paper uses dual-pooling method to obtain sequence average feature and maximum feature at the same time. Then we use the dual-pooled aspect/context vectors as the guide to learn the attention weight on context/aspect respectively so as to obtain the effective interactions that can determine the sentiment polarity of specific aspect.

Aspect2Context

This part first considers the global influence on context that calculates the attention weights of word-level feature and phrase-level feature in context. If the weight value is large, it indicates that the feature has closer semantic relationship with specific aspect. The average pooling value of a specific aspect is $Pool_{avg}^a = \frac{1}{M} \times \sum_{i=1}^{M} h_i^a$, and the maximum pooling value is $Pool_{\max}^a = \max(MHSA_{i,:}^a)$.

1) A-Aspect2Context learns to assign attention scores to the word-level feature $MHSA^c = [h_1^c, h_2^c, \ldots, h_N^c]$ and phrase-level feature $Tree\_MHSA^c = [h_1^{tree}, h_2^{tree}, \ldots, h_N^{tree}]$ of context with respect to the averaged aspect vector $Pool_{avg}^a$. For each vector h_i^c in contextual word-level feature and h_i^{tree} in contextual phrase-level feature, we can compute the attention score $\alpha_i^{avg\_ac} = \alpha_i(Pool_{avg}^a, h_i^c)$ and $\alpha_i^{avg\_tac} = \alpha_i(Pool_{avg}^a, h_i^{tree})$ according to formula 12.

Then the weighted combination of the contextual word-level output and phrase-level output is calculated according to formula 14 and 15 respectively:

$$m_{avg}^{ac} = \sum_{i=1}^{N} \alpha_i^{avg\_ac} \cdot MHSA_i^c \tag{14}$$

$$m_{avg}^{tac} = \sum_{i=1}^{N} \alpha_i^{avg\_tac} \cdot Tree\_MHSA_i^c \tag{15}$$

2)M-Aspect2Context learns to assign attention scores to the word-level feature $MHSA^c = [h_1^c, h_2^c, \ldots, h_N^c]$ and phrase-level feature $Tree\_MHSA^c = [h_1^{tree}, h_2^{tree}, \ldots, h_N^{tree}]$ of context with respect to the maximized aspect vector $Pool_{\max}^a$. For each vector h_i^c in contextual word-level feature and h_i^{tree} in contextual phrase-level feature, we can compute the attention score $\alpha_i^{\max\_ac} = \alpha_i(Pool_{\max}^a, h_i^c)$ and $\alpha_i^{\max\_tac} = \alpha_i(Pool_{\max}^a, h_i^{tree})$ according to formula 12.

Then the weighted combination of the contextual word-level output and phrase-level output is calculated according to the following formulas:

$$m_{\max}^{ac} = \sum_{i=1}^{N} \alpha_i^{\max\_ac} \cdot MHSA_i^c \tag{16}$$

$$m_{\max}^{tac} = \sum_{i=1}^{N} \alpha_i^{\max\_tac} \cdot Tree\_MHSA_i^c \tag{17}$$

Context2Aspect

This part mainly considers the global influence on aspects words, which has the similar learning process with *Aspect2Context*. We utilize the average pooling to obtain the averaged contextual word-level vector $Pool_{avg}^c = \frac{1}{N} \times \sum_{i=1}^{N} h_i^c$ and averaged contextual phrase-level vector $Pool_{avg}^{tc} = \frac{1}{N} \times \sum_{i=1}^{N} h_i^{tree}$. In addition, we utilize the maximum pooling to obtain the maximized contextual word-level vector $Pool_{\max}^c = \max(MHSA_{i,:}^c)$ and maximized contextual phrase-level vector $Pool_{\max}^{tc} = \max(Tree\_MHSA_{i,:}^c)$.

1)A-Context2Aspect learns to assign attention scores on aspects. For each word in the aspect phrase, we compute the attention score on both averaged word-level and averaged phrase-level context feature sequence. Then we obtain $\beta_i^{avg} = \alpha_i(Pool_{avg}^c, h_i^a)$ and $\beta_i^{tree\_avg} = \alpha_i(Pool_{avg}^{tc}, h_i^a)$ respectively according to formula 12. Combined with the hidden representations of specific aspects, we compute the weighted output as follows:

$$m_{avg}^{ca} = \sum_{i=1}^{M} \beta_i^{avg} \cdot MHSA^a \tag{18}$$

$$m_{avg}^{tca} = \sum_{i=1}^{M} \beta_i^{tree\_avg} \cdot MHSA^a \tag{19}$$

2)M-Context2Aspect also learns to assign attention scores on aspects. For each word in the aspect phrase, we compute

TABLE 2. Distributions of sentiment polarity categories for the three datasets.

Sentiment	Datasets					
	Laptop		Restaurant		Twitter	
	Train	Test	Train	Test	Train	Test
Positive	994	341	2164	728	1561	173
Negative	870	128	807	196	1560	173
Neutral	464	169	637	196	3127	346

the attention score on both maximized word-level and maximized phrase-level context feature sequence. Then we obtain $\beta_i^{\max} = \alpha_i(Pool^c_{\max}, h^a_i)$ and $\beta_i^{tree\_\max} = \alpha_i(Pool^{tc}_{\max}, h^a_i)$ respectively. Combined with the hidden representations of specific aspects, we compute the weighted output as follows:

$$m^{ca}_{\max} = \sum_{i=1}^{M} \beta_i^{\max} \cdot MHSA^a \tag{20}$$

$$m^{tca}_{\max} = \sum_{i=1}^{M} \beta_i^{tree\_\max} \cdot MHSA^a \tag{21}$$

E. OUTPUT LAYER

In the end, both the Aspect2Context and Context2Aspect attention vectors are concatenated as the final interaction representation $M \in R^{8d_h}$, and then we fed it into a softmax layer to determine the aspect sentiment polarity.

$$M = [m^{ac}_{avg}; m^{ac}_{\max}; m^{tac}_{avg}; m^{tac}_{\max}; m^{ca}_{avg}; m^{ca}_{\max}; m^{tca}_{avg}; m^{tca}_{\max}] \tag{22}$$

$$p = soft\max(W_p \times M + b_p) \tag{23}$$

where $W_p \in R^{C \times 8d_h}$ is the weight matrix and $b_p \in R^C$ is the bias. Set $C = 3$ as the number of categories of sentiment polarity.

The final loss function consists of the cross-entropy loss and L2 regularization, and the model is trained by minimizing the loss function that calculated as follows:

$$loss = -\sum_{i \in D}\sum_{j \in C} y_i^j \log p_i^j + \lambda \|\theta\|^2 \tag{24}$$

Here D is the number of training sets, $y_i \in R^c$ is the true sentiment polarity, and $p_i \in R^c$ is the predicted distribution of sentences to be classified, i is the index of a data sample, and j is the index of a sentiment class.

IV. EXPERIMENT AND ANALYSIS

A. DATASETS

In order to verify the effectiveness of the proposed model, we evaluate it on public datasets widely used in ABSA tasks: Laptop, Restaurant and Twitter. The first two datasets are from SemEval2014. The sentiment polarity of the above datasets is divided into positive, negative and neutral. Table 2 shows the details of the datasets.

B. EXPERIMENT SETTING

In the experiment, we set the initial word embeddings to 300-dimension Glove vectors [39] for all datasets. And for the words that out of vocabulary, we randomly sample their embeddings from the uniform distribution $U(-0.01, 0.01)$. The dimension of the hidden representation is set to 300. The head of the Transformer encoder and the Tree Transformer Encoder is set to 3. In order to avoid over fitting, the coefficient of L2 regular term is set to 0.001, and the random dropout rate is set to 0.5. The Adam optimizer with learning rate of 0.01 was used to train the model. The number of epoch was 10 and the batch size was 64.

C. BASELINE MODELS

In order to fully evaluate the performance of T-MGAN, we will compare it with the following 9 baseline approaches as comparisons, which are CNN, ATT-CNN, LSTM, TD-LSTM, AT-LSTM, ATAE-LSTM, IAN, IAD and MFIF.

- **CNN** [12] is a basic convolutional neural network model, which utilizes convolutional neural network to obtain the high-level features of the text to train the sentiment analysis model.
- **ATT-CNN** [17] integrates attention mechanism into convolutional neural network and trains the extracted high-level features into sentiment polarity classification model.
- **LSTM** [14] is a basic long short-term memory network. The hidden states of the last layer of the network is used as the final feature of the sentence and input them into the classifier for training.
- **TD-LSTM** [18] takes the target word as the center to divide the text into two parts, and inputs them into two LSTM networks in positive and reverse order respectively. Finally, it takes the last hidden states of LSTM networks for prediction.
- **ATT-LSTM** [36] integrates attention mechanism on the basis of LSTM and models the semantic relationship between the aspect target and context. It concatenates the output state of LSTM and aspect embedding to generate the attention weight value.
- **ATAE-LSTM** [36] also extends LSTM, which not only considers aspect embedding, but also considers the relationship between aspect and context word embedding. At the input level, the specific aspect word embedding is fused with each word embedding in the context to generate attention weight.
- **IAN** [36] employs two LSTM networks and attention mechanism to extract features from aspects and contexts, and then generates attention weights interactively as the final feature training model.
- **IAD** [38] employs the attention mechanism to obtain the important interactive information between aspect and context, and it is used for multi-level semantic classification.
- **MFIF** [2] takes word embedding and character embedding as input, and extracts aspect and context features interactively. And then uses GRU and attention mechanism to further obtain important features.
- **MGAN** [41] employs the fine-grained and coarse-grained attention mechanism to capture the interaction between aspect and context, and designs an aspect alignment loss to

depict the aspect-level interactions among the aspects that have the same context.

The above models are tested on semeval2014 and twitter datasets. The experimental results are shown in Table 3:

D. BASELINE MODEL ANALYSIS

Table 3 shows the performance comparison results of T-MGAN with other baseline methods, the bold in each column represents the item with the highest performance. In general, the T-MGAN model in this paper is better than other models in comparative experiments, which shows that the model is effective in ABSA tasks.

Comparing and analyzing the experimental results of these models, it can be observed that when there is no attention mechanism, the accuracy of CNN is lower than that of LSTM. This is because in the field of natural language processing, due to the complex syntactic structure and modifiers, the two adjacent words do not always have correlation. And CNN can only extract the features between a few words in the sentence, but cannot obtain the long-distance features between words, resulting in the CNN convolution operation cannot effectively obtain the information of the sentence. In addition, LSTM is superior to CNN in feature extraction of sequence data because of its gate mechanism can obtain long-term dependence and word order discrimination. Compared with the ordinary LSTM, TD-LSTM has a certain degree of improvement in the accuracy of the three datasets. This is because in TD-LSTM, the features of aspects and context are extracted by using the LSTM separately, and it models the left and right contexts of an aspect rather than the entire sentence. However, there is no weight distribution between each feature in the context and the aspect, that is, it cannot effectively pay attention to the contextual features that contribute more to the judgment of the emotional polarity of the specific aspect.

ATT-CNN and ATAE-LSTM are combined with attention mechanism on the basis of CNN and LSTM respectively. Compared with the original model, the accuracy of the laptop, restaurant and twitter datasets is improving approximately 2.79%, 0.96%, 1.15% and 2.20%, 2.90%, 3.38% respectively, which proves that attention mechanism can focus on learning the important information that in the task of ABSA. Comparing ATT-CNN and AT-LSTM, the accuracy of CNN combined with attention mechanism is lower than that of LSTM combined with attention mechanism, which is caused by the shortcomings of CNN model discussed in the above analysis in NLP task. ATAE-LSTM experimental results are better than AT-LSTM model, because the former further improves the input layer on the basis of the latter, as well as integrates the embedding of specific aspect words and each word in the context. Therefore, ATAE-LSTM makes the feature representation more comprehensive. Compared with ATAE-LSTM model, the experimental results of IAN model are improved. Because the model fully considers the feature relationship between aspects and context, it not only calculates the importance of each word in the context by pooling the aspect features with each word in the context, but also calculates the attention between the pooled context features and each word in the aspect which in order to learn the importance of each word in specific aspects. The IAD model also combines the attention mechanism in the LSTM, and considers the dependent information of different specific aspects in sentences with multiple specific aspects. The experimental results are further improved compared with the IAN model. In addition to considering the interaction characteristics between a particular aspect and context, the MFIF model also carries out multi-feature processing in the input part and adds character characteristics, thus obtaining better experimental results. The MGAN model employs not only the coarse-grained attention mechanism but also the proposed fine-grained attention mechanism to obtain the interaction between aspect and context. And designs an aspect alignment loss to depict the aspect-level interactions among the aspects that have the same context. This model has the best performance among all the other comparison models. The above model demonstrates the important role of attention mechanisms, the importance of capturing the interaction characteristics between a particular aspect and context, and the advantages of multiple feature representations.

The proposed model T-MGAN outperforms baseline methods in most cases because it adopts the transformer structure, which can obtain the aspect features and context features from different angles under multiple different linear transformations. In addition, the tree transformer structure is also used in the context feature acquisition part to obtain the phrase-level global features containing structural information from different perspectives, which makes the model have advantages in feature representation. Moreover, when using the attention mechanism to obtain the interaction features between the aspects and the context, the dual-pooling method is adopted, which combines the global features of the dual-pooling with the local features of the aspects, so as to learn and express the important information between the aspect and the context in a deeper level. However, on the twitter dataset, the experimental result of T-MGAN is poorer than that of the MGAN. One of the reason may be that the expression form of the dataset is more colloquial and there is no standard expression, which leads to the model has no obvious advantage in obtaining phrase-level features. On the other hand, the restaurant and laptop datasets belong to specific domains while the twitter dataset covers a general domain with larger vocabulary. There are some words that out of vocabulary in word embedding for the Twitter dataset. Generally speaking, the quality of lexical and syntactic features in this model is high, and the interactive feature acquisition method between aspect and context is effective in ABSA tasks.

E. ANALYSIS OF T-MGAN MODEL

In this section, we mainly explore the effect of the number of layers of word-level features obtained from Transformer Encoder and the number of layers of phrase-level features obtained by Tree Transformer Encoder on the performance of the T-MGAN model.

TABLE 3. The performance of different models. The results with symbol "†" are our reimplemented, and others baseline results are retrieved from the original papers. The results with '–' means not reported in the original papers, and the best performance is marked in bold.

Model	Laptop		Restaurant		Twitter	
	Accuracy(%)	Macro-F1(%)	Accuracy(%)	Macro-F1(%)	Accuracy(%)	Macro-F1(%)
CNN	59.32†	55.01†	67.23†	59.21†	68.03†	59.75†
ATT-CNN	62.11†	58.64†	68.19†	60.35†	69.18†	63.10†
LSTM	66.50†	61.41†	74.30†	65.32†	66.50†	62.45†
ATT-LSTM	68.90	-	76.60	-	-	-
ATAE-LSTM	68.70	-	77.20	-	69.88	-
TD-LSTM	71.83	68.43	78.00	66.73	66.62	64.01
IAN	72.10	67.83†	78.60	69.65†	70.42†	68.03†
IAD	72.50	-	79.00	-	-	-
MFIF	75.34	-	80.35	-	71.90	-
MGAN	75.39	72.47	81.25	71.94	**72.54**	**70.81**
T-MGAN（our model）	**76.38**	**73.02**	**82.06**	**72.65**	71.23	70.63

1) EFFECT OF THE NUMBER OF TRANSFORMER ENCODER'S LAYERS

Transformer Encoder is composed of N Layers. Choosing a different number of layers will affect the ability to acquire features. In order to determine the optimal number of layers in Transformer Encoder, we first use the Tree Transformer Encoder for phrase-level feature extraction in the context of the T-MGAN model. Part of it is omitted, and only the Transformer Encoder for extracting aspects and context features is retained. The experimental verification is performed on the number of layers 1-6 in turn, and the result is shown in Figure 5. The rules presented by the experimental results are basically consistent with our experience. When the number of layers is too small, the feature extraction of the model is not perfect. Therefore, as the number of layer increases, the effective features learned by the model increase, and the accuracy of the experiment gradually increases. When the number exceeds a certain number, too many model parameters make the model have the risk of overfitting, and the experimental accuracy is reduced. When the number of layers is 3, the laptop and restaurant datasets have the highest accuracy rates, which are 73.15% and 80.10%, respectively. When the number of layers is 4, the accuracy of the Twitter dataset is 70.03%, which is 0.07% higher than the result of the number of layers of 3. But when the number of layers is 3, the accuracy of the other two datasets is improved much morethan 4-layers, so we choose the number of Transformer Encoder layer as 3.

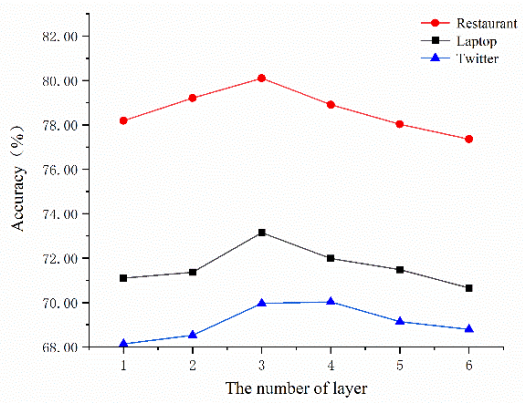

FIGURE 5. The performance of different number of layers of Transformer Encoder.

2) EFFECT OF THE NUMBER OF TREE TRANSFORMER ENCODER'S LAYERS

Tree Transformer Encoder also has a multi-layered structure. Each level will be combined with some sub-phrases, and several shorter phrase components will be gradually added from the lower level to the higher level, and then the contextual phrase-level features will be learned through this module. In this section of the experiment, the number of layers of Transformer Encoder module in the T-MGAN model is set to 3, and the number of layers of Tree Transformer Encoder module is set to 1-6 to conduct experiments in sequence, and the results are shown in Figure 6. It can be seen from the experimental results that when the number of layer is 1, the experimental result is slightly higher than the result without adding the Tree Transformer Encoder module, indicating that the module can indeed learn some other important features. When the number of layers is 3, the model has learned more sufficient phrase-level features, which further improves the accuracy of the model and reaches the highest accuracy of the model.

3) VERIFY THE EFFECTIVENESS OF IMPORTANT MODULE

In this section, we list the variants of T-MGAN model, which are used to analyze the effects of Transformer Encoder module, Tree Transformer Encoder module and dual-pooling method, respectively

•**T-MGAN w/o average pooling**uses Transformer Encoder module and Tree Transformer Encoder module in intra-feature extraction layer, and uses only maximum pooling in inter-feature multi-attention layer. Average pooling are not used.

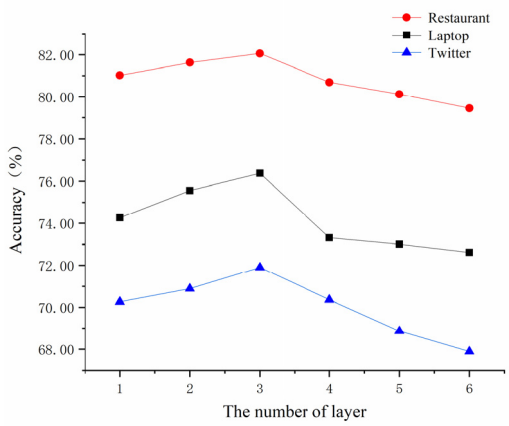

FIGURE 6. The performance of different number of layers of Tree Transformer Encoder.

TABLE 4. Performance of important modules in T-MGAN.

Method	Accuracy(%)		
	Laptop	Restaurant	Twitter
T-MGAN w/o average pooling	75.72	81.61	70.29
T-MGAN w/o maximum pooling	75.61	81.24	70.43
T-MGAN w/o Tree Transformer Encoder	73.15	80.10	69.96
T-MGAN	**76.38**	**82.06**	**71.23**

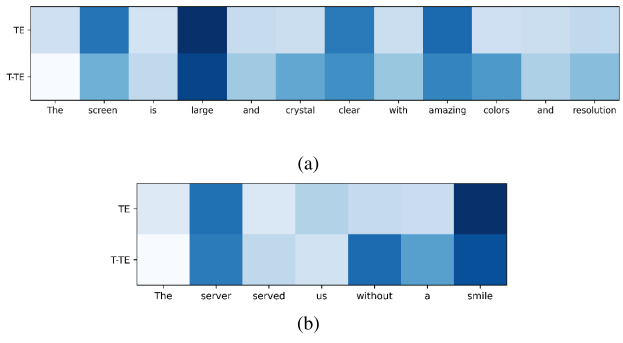

FIGURE 7. The attention visualizations on aspect:(a) aspect word is "*screen*";(b) aspect word is "*server*".

•**T-MGAN w/o maximum pooling**uses Transformer Encoder module and Tree Transformer Encoder module in intra-feature extraction layer, and uses only average pooling in inter-feature multi-attention layer. Maximum pooling are not used.

•**T-MGAN w/o Tree Transformer Encoder**uses only Transformer Encoder module in intra-feature extraction layer. In addition, this model uses both maximum pooling and average pooling in inter-feature multi-attention layer. Tree Transformer Encoder module is not used.

Table 4 shows the results of the important modules in T-MGAN and we can observe that T-MGAN performs better than the other methods. Comparative analysis of **"T-MGAN w/o Tree Transformer Encoder"** and **"T-MGAN"**, the latter performs better than the former one on both the dataset, which demonstrates that Tree Transformer Encoder module can indeed learn more sufficient representations than the model only equipped with Transformer Encoder module. **"T-MGAN"** performs better than both **"T-MGAN w/o average pooling"** and**"T-MGAN w/o and maximum pooling"**, this result proves that the dual-pooling method used in this paper is better than the single-pooling method.

4) CASE ANALYSIS

To further evaluate the performance of the proposed model, two sentences were randomly selected from the Laptop and Restaurant datasets to visualize their attention results. And we utilize TE and T-TE to represent the distribution of attention output after encoding by Transformer Encoder and Tree Transformer Encoder respectively. As shown in Figure 7, color intensity represents the strength of the attention weights. The first sentence is:

"*The screen is large and crystal clear with amazing colors and resolution.*"

Figure 7(a) shows the attention weights of the aspect word "screen". TE mainly focuses on the directly related adjectives in the sentence, such as "*large*" and "*clear*". The focus of T-TE is different from TE. It is more inclined to focus on the phrases "*crystal clear*" and "*amazing colors*". These phrases are the key phrases to determine the emotional polarity of the aspect word "screen", but the attention to a certain word is not as high as TE. The second sentence is:

"*The server served us without a smile.*"

Figure 7(b) shows the attention weights of the word "*server*" in a specific aspect. It can be seen that TE mainly focuses on the adjective "*smile*" related to "*server*" in the sentence, but this word has a positive sentiment tendency. Focusing only on the word may lead to bias in the judgment of emotional polarity. However, T-TE has a higher attention weight for the phrase "*without a smile*", which can effectively reflect the importance of the negative word "*without*" and learn the characteristics of sentiment polarity reversal. Through the analysis of the above visualization results, the T-MGAN model can pay attention to the important word-level features related to the word meaning of aspects and the phrase-level features related to the word polarity of the specific aspect, which verifies that the model has good sentiment semantic extraction ability.

V. CONCLUSION

This paper proposes a T-MGAN model for aspect-based sentiment analysis. We combine Transformer Encoder and Tree Transformer Encoder with specific sub-tasks, utilize T-MGAN to model the word-level and phrase-level features of aspects and contexts. T-MGAN not only acquires semantic features, but also effectively obtains language hierarchy and syntax information. In addition, the dual-pooling method is used to extract the key features in the hidden layer features of aspect and context, and the attention mechanism is used multiple times to effectively obtain the fine-grained associated emotional features between the specific aspect and the context. Through the attention weight visualization, it is

verified that the Transformer Encoder module in this model can effectively focus on the important words in the sentence, while the Tree Transformer Encoder module can learn syntactic information to effectively focus on important phrases. A comparative analysis with other models verifies that the accuracy of this model is better than the comparison model. In future research, we will deeply study the more colloquial texts on social platforms such as Twitter, and propose a more suitable model structure for this type of text. In addition, we are considering to build a civil aviation dataset, and we will try to verify the effectiveness of the proposed model on it.

REFERENCES

[1] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*. New York, NY, USA: ACM, 2004, pp. 168–177.

[2] B. Zeng, X. Han, F. Zeng, R. Xu, and H. Yang, "Multifeature interactive fusion model for aspect-based sentiment analysis," *Math. Problems Eng.*, vol. 2019, pp. 1–8, Dec. 2019, doi: 10.1155/2019/1365724.

[3] Z. Kastrati, A. S. Imran, and A. Kurti, "Weakly supervised framework for aspect-based sentiment analysis on students' reviews of MOOCs," *IEEE Access*, vol. 8, pp. 106799–106810, 2020, doi: 10.1109/ACCESS.2020.3000739.

[4] Y. Han, M. Liu, and W. Jing, "Aspect-level drug reviews sentiment analysis based on double BiGRU and knowledge transfer," *IEEE Access*, vol. 8, pp. 21314–21325, 2020, doi: 10.1109/ACCESS.2020.2969473.

[5] I. Sindhu, S. M. Daudpota, K. Badar, M. Bakhtyar, J. Baber, and M. Nurunnabi, "Aspect-based opinion mining on student's feedback for faculty teaching performance evaluation," *IEEE Access*, vol. 7, pp. 108729–108741, 2019.

[6] Z. Kastrati, B. Arifaj, A. Lubishtani, F. Gashi, and E. Nishliu, "Aspect-based opinion mining of students' reviews on online courses," in *Proc. 6th Int. Conf. Comput. Artif. Intell.*, 2020, pp. 510–514.

[7] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Edinburgh, U.K., 2011, pp. 151–161.

[8] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent Twitter sentiment classification," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, vol. 2, 2014, pp. 49–54. [Online]. Available: http://aclweb.org/anthology/P14-2009

[9] T. H. Nguyen and K. Shirai, "PhraseRNN: Phrase recursive neural network for aspect-based sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Lisbon, Portugal, 2015, pp. 2509–2514. [Online]. Available: http://aclweb.org/anthology/D/D15/D15-1298.pdf

[10] C. R. Aydin and T. Gungor, "Combination of recursive and recurrent neural networks for aspect-based sentiment analysis using inter-aspect relations," *IEEE Access*, vol. 8, pp. 77820–77832, 2020, doi: 10.1109/ACCESS.2020.2990306.

[11] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2014, pp. 655–665. [Online]. Available: https://www.aclweb.org/anthology/P14-1062/

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

[13] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for text classification," 2016, *arXiv:1606.01781*. [Online]. Available: http://arxiv.org/abs/1606.01781

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[15] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *CoRR*, 2014. [Online]. Available: http://arxiv.org/abs/1412.3555

[16] X. Li, L. Bing, W. Lam, and B. Shi, "Transformation networks for target-oriented sentiment classification," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2018, pp. 946–956. [Online]. Available: http://aclweb.org/anthology/P18-1087

[17] L. Wang, Z. Cao, G. de Melo, and Z. Liu, "Relation classification via multi-level attention CNNs," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2016, pp. 1298–1307.

[18] D. Tang, B. Qin, X. Feng, and T. Liu, "Effective LSTMs for target-dependent sentiment classification," in *Proc. 26th Int. Conf. Comput. Linguistics*, 2016, pp. 3298–3307.

[19] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent attention network on memory for aspect sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 452–461. [Online]. Available: http://aclweb.org/anthology/D17-1047

[20] 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.*, Long Beach, CA, USA, 2017, pp. 5998–6008. [Online]. Available: http://papers.nips.cc/paper/7181-attention-is-all-youneed

[21] Y. S. Wang, H. Y. Lee, and Y. N. Chen, "Tree transformer: Integrating tree structures into self-attention," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2019, pp. 1060–1070. [Online]. Available: https://arxiv.org/abs/1909.06639

[22] E. Cambria, "Affective computing and sentiment analysis," *IEEE Intell. Syst.*, vol. 31, no. 2, pp. 102–107, Mar. 2016. [Online]. Available: http://ieeexplore.ieee.org/document/7435182/

[23] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retr.*, vol. 2, nos. 1–2, pp. 1–135, 2008.

[24] B. Liu, "Sentiment analysis and opinion mining," *Synthesis Lectures Hum. Lang. Technol.*, vol. 5, no. 1, pp. 1–167, 2012. [Online]. Available: http://www.morganclaypool.com/doi/abs/10.2200

[25] K. Li, C. Chen, X. Quan, Q. Ling, and Y. Song, "Conditional augmentation for aspect term extraction via masked sequence-to-sequence generation," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 7056–7066.

[26] M. Dragoni, M. Federici, and A. Rexha, "An unsupervised aspect extraction strategy for monitoring real-time reviews stream," *Inf. Process. Manage.*, vol. 56, no. 3, pp. 1103–1118, May 2019.

[27] X. Ding, B. Liu, and P. S. Yu, "A holistic lexicon-based approach to opinion mining," in *Proc. Int. Conf. Web Search Web Data Mining (WSDM)*, Palo Alto, CA, USA, 2008, pp. 231–240, doi: 10.1145/1341531.1341561.

[28] S. Kiritchenko, X. Zhu, C. Cherry, and S. Mohammad, "NRC-Canada-2014: Detecting aspects and sentiment in customer reviews," in *Proc. 8th Int. Workshop Semantic Eval. (SemEval)*, Dublin, Ireland, 2014, pp. 437–442. [Online]. Available: http://aclweb.org/anthology/S/S14/S142076.pdf

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

[30] W. Xue and T. Li, "Aspect based sentiment analysis with gated convolutional networks," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2018, pp. 2514–2523.

[31] D. Tang, B. Qin, and T. Liu, "Aspect level sentiment classification with deep memory network," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 214–224.

[32] S. Ruder, P. Ghaffari, and J. G. Breslin, "A hierarchical model of reviews for aspect-based sentiment analysis," 2016, *arXiv:1609.02745*. [Online]. Available: http://arxiv.org/abs/1609.02745

[33] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent attention network on memory for aspect sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 463–472.

[34] D.-T. Vo and Y. Zhang, "Target-dependent twitter sentiment classification with rich automatic features," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, Buenos Aires, Argentina, Jul. 2015, pp. 1347–1353.

[35] B. Zeng, H. Yang, R. Xu, W. Zhou, and X. Han, "LCF: A local context focus mechanism for aspect-based sentiment classification," *Appl. Sci.*, vol. 9, no. 16, p. 3389, Aug. 2019, doi: 10.3390/app9163389.

[36] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 606–615. [Online]. Available: http://aclweb.org/anthology/D16-1058

[37] D. Ma, S. Li, X. Zhang, and H. Wang, "Interactive attention networks for aspect-level sentiment classification," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Melbourne, VIC, Australia, 2017, pp. 4068–4074.

[38] D. Hazarika, S. Poria, P. Vij, G. Krishnamurthy, E. Cambria, and R. Zimmermann, "Modeling inter-aspect dependencies for aspect-based sentiment analysis," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 2, 2018, pp. 266–270.

[39] M. Jiang, J. Wu, X. Shi, and M. Zhang, "Transformer based memory network for sentiment analysis of Web comments," *IEEE Access*, vol. 7, pp. 179942–179953, 2019, doi: 10.1109/ACCESS.2019.2957192.
[40] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543, doi: 10.3115/v1/D14-1162.
[41] F. Fan, Y. Feng, and D. Zhao, "Multi-grained attention network for aspect-level sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 3433–3442.

JIAHUI SUN received the M.S. degree in information and communication engineering from the Civil Aviation University of China, Tianjin, China, in 2019. She is currently an Assistant with the Basic Experiment Center, Civil Aviation University of China. Her research interests include artificial intelligence and natural language processing.

PING HAN received the M.S. and Ph.D. degrees from Tianjin University, Tianjin, China, in 1989 and 2004, respectively. She is currently a Professor with the College of Electronic Information and Automation, Civil Aviation University of China. Her current research interests include signal processing and pattern recognition and SAR and PolSAR image interpretation.

ZHENG CHENG received the M.S. degree in electronic and communication engineering from the Civil Aviation University of China, Tianjin, China, in 2017. He is currently an Experimentalist with the Basic Experiment Center, Civil Aviation University of China. His current research interests include image processing, pattern recognition, and deep learning.

ENMING WU was born in Shandong, China, in 1985. He received the master's degree in detection technology and automation equipment from the Tianjin University of Science and Technology, in 2011. He is currently the Director of electronic technology innovation and entrepreneurship with the Civil Aviation University of China. His main research interests include UAV formation flight control and artificial intelligence control.

WENQING WANG received the B.S. degree in information and communication engineering from the Civil Aviation University of China, where he is currently pursuing the M.S. degree. His research interests include deep learning and trajectory prediction.