Applied Intelligence (2022) 52:5867–5879
https://doi.org/10.1007/s10489-021-02724-5

# Cooperative gating network based on a single BERT encoder for aspect term sentiment analysis

Yuqing Peng[1] · Tengfei Xiao[1] · Hongtao Yuan[1]

Accepted: 27 July 2021 / Published online: 23 August 2021


**Abstract**
In recent years, BERT encoder methods have been widely used in aspect term sentiment analysis (ATSA) tasks. Many ways of putting text and aspect term into a BERT sentence encoder separately aim to create vectors by obtaining context and aspect words. However, the semantic relevance of these initially extracted context-hiding vectors and aspect word-hiding vectors is poor. Moreover, they are easily affected by irrelevant words. Therefore, the CGBN model is proposed in this paper, which uses only the sentence sequence as the input to the BERT encoder. Moreover, the context-hiding vectors and aspect word-hiding vectors containing rich semantic association information were able to be extracted simultaneously for the first time. In addition, this paper proposes a new interactive gating mechanism called a co-gate. Compared with the general interactive feature extraction mechanism, it can not only effectively reduce the interference of noisy words but also fuse the information of context and aspect term better and capture emotional semantic features. To enhance the ability of BERT to be fine-tuned with domain data, the pretraining file of BERT Post Training (BERT-PT) is used in this paper to fine-tune the CGBN model. A method of domain adaptation is also applied with combined training sets, thus enhancing the training effect of the target domain data. Experiments and analysis prove the validity of the model.

**Keywords** Aspect term sentiment analysis · BERT encoder · Co-gate mechanism · Domain adaptation

## 1 Introduction

With the rapid development of social networks, e-commerce and other online media, people are increasingly expressing their opinions on services or products through online channels such as blogs and forums. User-generated comment data show an exponential growth trend, which not only greatly improves the decision-making quality of organizations and individuals but also greatly increases the quality of services and products. [1–4]. To make full use of these extensive information resources, emotion analysis has become an important tool for extracting human emotion information [5].

Aspect-based sentiment analysis (ABSA) is a fine-tuning task within sentiment analysis that aims to predict the sentiment polarity of target entities or aspect categories in text. ABSA can be divided into four subtasks: (1) Opinion target extraction (OTE)('Opinion target' is synonymous with 'aspect term'),(2) Aspect category detection (ACD),(3) Aspect Term Sentiment Analysis (ATSA), and(4) Aspect Category Sentiment Analysis (ACSA) [5, 6]. OTE and ACD detect a target entity in a sentence and its corresponding aspect category, respectively. ATSA determines the emotional polarity of a target entity, and ACSA predicts the emotional polarity of a related aspect category. For example, in the sentence fragment "but the staff was so horrible to us", ATSA judges the emotional polarity of the target entity "staff", and ACSA predicts the polarity of the aspect category "service" that corresponds to the sentence, even if "service" does not appear in the sentence [6]. This paper focuses on the ATSA task.

In recent years, deep learning, which can capture the syntactic and semantic features of text, has been widely used in ATSA tasks without the need for advanced feature engineering. The traditional neural network method uses GloVe [7] or Word2Vec [8] to obtain the word embedding vector of each token in the sentence sequence. It constructs task-related feature extraction layers through recurrent neural networks and attention mechanisms. Recently, a model based on the BERT [9] structure has achieved excellent results performing the ATSA task. BERT, a deep bidirectional encoder structure

✉ Yuqing Peng
pengyuqing@scse.hebut.edu.cn

[1] School of Computer Science and Engineering, Hebei University of Technology, Tianjin, China

based on Transformer [9, 10], avoids the need for task-specific architectures by using pretraining.

However, there are some problems in these methods at present. The first problem is the poor relevance of the initially extracted context-hiding vectors and aspect word-hiding vectors. The traditional methods of using context-independent word embedding vectors are not enough to capture the complex semantic dependencies in sentences, which will bring performance bottlenecks [11]. Although most of the models use the BERT encoder to capture the contextual representation vector and the aspect term representation vectors, their associated information is still insufficient, especially for the aspect term. In addition, in the face of sentences containing multiple aspect terms, the contextual representation vector may not be able to accurately distinguish the emotional polarity corresponding to a specific aspect term. Therefore, better results can be obtained by more complex interactive feature extraction. Another problem is that many models based on the attention mechanism use the average pooling vector of an aspect term to extract context features when extracting the relationship between the context and the aspect term. However, the pooling vector will inevitably bring in interference words if the aspect term is composed of multiple words. For example, in the target phrase "size of the screen", "of" is an interference word. In addition, the aspect term may be composed of multiple entities that have differing influences on sentiment classification. In the above example, "size" is more important than "screen" [12]. To overcome these shortcomings and better capture the semantic relationship between context and aspect term, attention-based models have become increasingly complex and computationally expensive. The third problem is that the ATSA task is a fine-grained task that requires high corpus labeling. However, the training set for ATSA is so small that its model may not be able to complete sufficient training, limiting its performance. For example, the SemEval 2014 dataset is a benchmark dataset for ATSA, but its training sets in the restaurant field and the laptop field have only approximately 3600 and 2300 data points, respectively.

This paper addresses the problems above. First, the contextual representation vector and aspect term representation vectors extracted by two BERT encoders have poor interactive feature capability. To effectively overcome this, the CGBN model uses only the sentence sequence as the input to its BERT encoder. Next, the context and aspect word representation vectors that contain rich semantic information are simultaneously extracted from the last layer of the BERT encoder. Second, a new nonparallel interactive gating mechanism co-gate (cooperative gating mechanism) is proposed in this paper to capture the relationship between the context and the aspect term more effectively. The shortcomings of traditional interaction feature extraction can be avoided, and the high computational cost of the attention mechanism [13] can be reduced. The gating mechanism fuses context information into the aspect term representation vector and then computes the context representation vector based on the specific aspect term. It reflects the relevance of given aspect words as well as context vectors in each dimension. Irrelevant emotional features can also be blocked and filtered out. Third, this paper fine-tunes its model with BERT-PT. BERT-PT applies a domain-specific corpus to pretraining tasks, which is useful for injecting domain knowledge and reducing the "domain bias" of the BERT benchmark pretraining corpus. Moreover, the model can grasp domain-related features better [14]. Additionally, the domain adaptability of the CGBN model is explored in this paper. The model is tested on multiple training sets to assess the impact of different training sets on its performance. It is found that the fusion of restaurant and laptop training data can greatly enhance the performance of these two target areas. To optimize its cross-domain performance, the model adaptively adjusts its parameters according to the target domain so that it can obtain more sufficient training. We also find the reasons why the integration of the twitter training set has a poor effect. The details can be seen in Section 4.4.2. Finally, the performances of the model on the restaurant dataset, laptop dataset, and ACL-twitter dataset are evaluated [15].

The main contributions of this paper are as follows.

(1) We use only sentence sequences as the input to the BERT encoder and then extract the hidden vectors of the context and aspect words in the encoder at the same time.
(2) We propose a new interactive gating mechanism co-gate, which enhances the interactive feature extraction capabilities of our model.
(3) We explore the influence of different combinations of training sets and the BERT-PT pretraining parameters on model performance.

## 2 Related work

In recent years, deep learning has shown excellent performance in various NLP tasks and is also used in ATSA [16]. Recurrent neural networks (RNNs) have been widely applied in ATSA tasks [17–21]. These models adopt RNNs for feature encoding of context and aspect. Although RNN is good at dealing with sequence problems, it has poor parallelization and cannot handle the long-term dependence of complex sentences. It may also cause gradient disappearance and gradient explosion problems. To solve these problems, researchers introduced the attention mechanism [10] which calculates the semantic correlation between each word in a sentence so that each word carries global semantic information. Therefore, the attention mechanism is also widely used in

ATSA [12, 21–26]. For example, the CAMN model proposed by Lv et al. [21] introduced a multilayered multi-head attention (MHA) mechanism to continuously update the aspect vector. The RAO-CNN model proposed by Wu et al. [22] used a residual attention mechanism to reduce the problem of losing original information in the original attention mechanism. The coattention-LSTM model raised by Yang et al. [12] adopted a nonparallel interactive attention mechanism that calculated aspect representation vectors through the attention mechanism. In addition, it also made full use of the key information in the target word to study context representation.

In addition to the RNN and attention mechanisms, a gating mechanism reflecting the relevance of given aspect words and context vectors in each dimension [17] is also widely used in ATSA. The mechanism can not only regulate information flow and process context information but also block and filter out irrelevant emotional features. Kai et al. [17] developed an FDN model that applied a dual-gating mechanism to accurately distinguish the emotional characteristics of the context belonging to different aspect terms. Li et al. [27] proposed the GBCN model which combined context-aware aspect embedding vectors with the context representation vectors extracted from BERT with the help of a gating mechanism. Avinash Kumar et al. [28] adopted interactive gating to model aspect words and context.

There have been many recent studies that associate feature engineering (such as dependency trees, POS tags, and external knowledge) with ATSA. For example, the CDT model presented by Sun et al. [29] and the ASGCN model proposed by Zhang et al. [30] enhanced sentence representation vectors by acting on the graph convolutional network (GCN) of a sentence dependency tree. The R-GAT model proposed by Wang et al. [19] constructed a dependency tree with aspect words as the root nodes and dropped unnecessary relationships. Shuang et al. [20] believed that adjectives, adverbs, and verbs contained important information and regarded POS information as the basis for calculating attention weights. Chen et al. [31] incorporated the external knowledge of an emotional knowledge graph into their model to alleviate its poor performance when using a small training corpus.

Traditional deep learning models rely on context-free, static GloVe or Word2Vec word embedding vectors. These vectors cannot model polysemy words, limiting the performance of the model. In the past two years, pretraining models such as ELMo [32], GPT [33], BERT [9], and XLNET [34] have achieved excellent ATSA task results, and the models that are based on a BERT structure have been widely used. The application of BERT takes three forms. The first form uses BERT to obtain the embedding vectors of context and aspect words. For example, the attention coding network AEN-BERT proposed by Song et al. [26] extracted context-hiding vectors and aspect word-hiding vectors by using two BERT encoders. The relationship between them was obtained through the attention mechanism. Zhang et al. [24] proposed the interactive multi-head attention network IMAN. Using an attention mechanism and convolution, IMAN dealt with the representation vectors of context and aspect words obtained from BERT. Another BERT application form transforms ABSA into a sentence pair classification task. For example, the BERT-SPC model of Song et al. [26] uses sentence pairs composed of context and aspect words as the input to the BERT encoder and then extracts the [CLS] token corresponding to a hidden vector as its final vector representation. The third form of BERT application enhances its domain adaptability in the pretraining phase. Xu et al. [14] and Alexander Rietzler et al. [35] both held that BERT lacked domain knowledge and task-related knowledge, so they applied a domain-specific corpus at the BERT pretraining stage. The use of these task-related BERT pretraining models can greatly improve performance. The first and third forms of BERT applications are the most widely used and are the subjects of this article.

The domain adaptation method is not only applied to the BERT pretraining stage but also has a wide range of applications in ABSA. Some researchers introduced domain adaptation methods into the ABSA domain. For example, Rana et al. [36] used a dictionary containing many concepts and aspect words related to domains, thus improving the ability of aspect extraction tasks. In addition, Alexander Rietzler et al. proposed three combinations of training sets: using the same domain training set (in-domain training), using different domain training sets (cross-domain training) and combining multiple domain training sets (joint-domain training). These different training set combinations enhanced the model's feature learning ability in the BERT fine-tuning stage.

In general, a model can achieve better results if it is based on a pretraining model combined with a gating or attention mechanism and other structures.

## 3 Proposed methodology

The CGBN model architecture is shown in Fig. 1 and consists of four layers: an input layer, a BERT encoder layer, a co-gate layer, and an output layer. Different from the traditional model in obtaining context and aspect encoding vectors, the single BERT encoder used by the CGBN model is lighter and its extracted context and aspect encoding vectors contain richer semantic information. These two extraction methods are shown in Figs. 2 and 3. The hidden context and aspect vectors are fed into the co-gate layer to obtain their interactive representation vectors. The context and aspect representation vectors are combined in the output layer and then sentiment classification is performed. To enhance the fine-tuning ability of BERT for downstream tasks and alleviate the problem of mismatch between the pretraining and fine-tuning stages of the

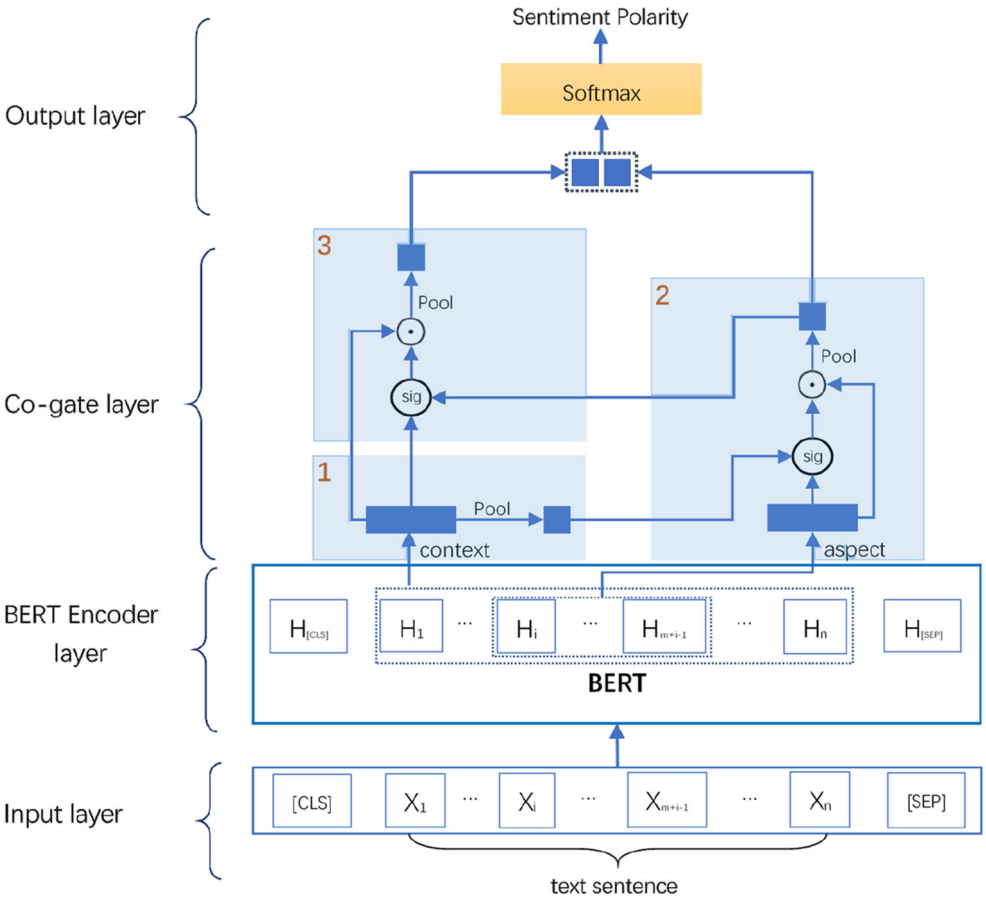


**Fig. 1** Structure of the CGBN model

BERT training data, this paper uses the BERT-PT pretraining parameters to fine-tune the CGBN model and uses fused training data to enhance the target field training.

### 3.1 Task definition

Given a review sentence $W_c = \{X_1, X_2, \ldots, X_n\}$ containing $n$ words and the aspect term sequence $W_a = \{X_i, \ldots, X_{m+i-1}\}$ containing $m$ words, the goal of ATSA is to predict the emotional polarity $y$ corresponding to the aspect term $W_a$ in the review sentence $W_c$, where $y \in \{positive, neutral, negative\}$.

### 3.2 Input layer

In this paper, a sentence sequence is used as the input to a BERT encoder. Inspired by Gao et al. [35], an aspect embedding vector is obtained from the hidden sequence after BERT encoding, avoiding separate encoding of the aspect.

Next, the sentence token sequence is converted into the format required by the BERT encoder. BERT uses the [*CLS*] token as the start tag of the sequence and the [*SEP*] token as the end tag of the sequence. The token sequence entered is as follows: [*CLS*] + text sequence +[*SEP*].

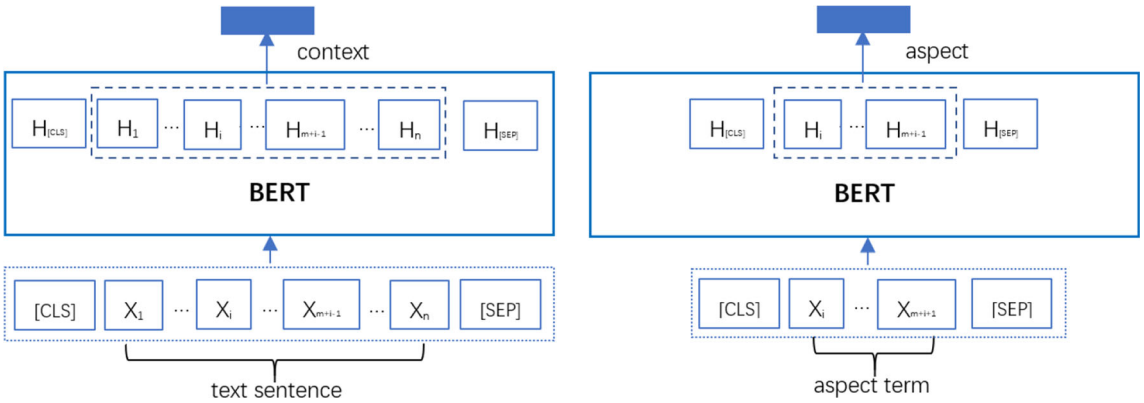


**Fig. 2** Traditional models use two BERT encoders to extract context and aspect encoding vectors

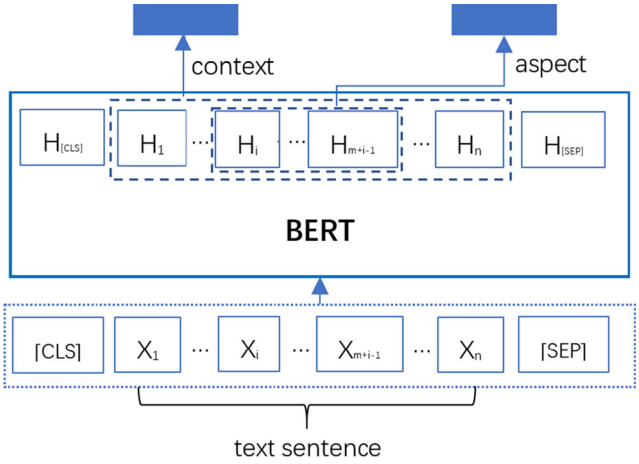


**Fig. 3** Only sentence text is used as the input to the BERT Encoder, and context and aspect encoding vectors are extracted

For the input token sequence, the input representation is the combination of the token embedding vector, segment embedding vector, and position embedding vector.

### 3.3 BERT encoder layer

Many models use two BERT encoders to encode context and aspect words [24, 26]. However, the correlation between these obtained context and aspect coding vectors is very poor. A complex interactive feature extraction layer is needed to further obtain their interactive relationship. The TD-BERT model of Gao et al. [35] extracts only the aspect vector from the context hidden sequence as the representation vector and achieves good results. Inspired by this study, our CGBN model extracts both the context hidden vector $H^c$ and the aspect hidden vector $H^a$ with a BERT encoder for the first time. Compared with traditional methods, the extracted $H^c$ and $H^a$ contain rich semantically related information. The calculation of $H^c$ and $H^a$ is shown in Eq. (1):

$$H^c, H^a = BERT(S) \tag{1}$$

where $S$ represents a sentence token sequence with [*CLS*] and [*SEP*] tags added at the beginning and end. *BERT*() represents the model structure after the pretraining phase. We need to input text sequences into *BERT*() for fine-tuning.

### 3.4 Co-gate layer

This section will introduce how the co-gate layer calculates the interactive characteristics of aspect and context. Co-gate is a nonparallel interactive gating mechanism. It obtains the context initial pooling vector $c_{pool}^{init}$, calculates the aspect term representation vector $a_{pool}^{'}$ according to $c_{pool}^{init}$, and finally calculates the context representation vector $c_{pool}^{'}$ according to $a_{pool}^{'}$. Compared with the general parallel-structured interactive gating mechanism, the co-gate mechanism effectively reduces the problem of introducing noise words when calculating the context representation vector from a specific aspect term.

#### 3.4.1 Getting the context initial vector

To integrate the context information into the aspect term representation vector, the initial representation vector of the context needs to be obtained. As shown in Eq. (2), this paper first calculates the average pooling vector $c_{pool}$ of the context, then uses the *Tanh*() activation function to adjust $c_{pool}$ adaptively to make it more effective in combination with the aspect term hidden vector $H^a$. The obtained context initial vector $c_{pool}^{init}$ is shown in Eq. (3):

$$c_{pool} = \sum_{i=1}^{n} H_i^c / n \tag{2}$$

$$c_{pool}^{init} = Tanh\left(W_1 c_{pool} + b_1\right) \tag{3}$$

where *Tanh*() is the activation function, $W_1 \in R^{d_H \times d_H}$ and $b_1 \in R^{d_H}$ are learnable parameters, and $R^{d_H}$ is the dimension of the BERT hidden layer.

#### 3.4.2 Get aspect term representation vector

The aspect term may not contain sentiment features. It is not enough to capture emotional features by using only the aspect pooling vector to represent the aspect term representation vector. Therefore, contextual information needs to be incorporated. This paper uses *sigmoid* gating to calculate the correlation weight $W^a$ between each word in the aspect term and the context initial vector $c_{pool}^{init}$ in each dimension. $W^a$ is mapped to the (0, 1) interval, which reflects the importance of different words in the aspect term for sentiment classification. The calculation is shown in Eq. (4):

$$W_j^a = Sigmoid\left(W_2 H_j^a + W_3 c_{pool}^{init} + b_2\right) \tag{4}$$

where $H_j^a \in R^{d_H}$ is the hidden vector of the $j$th word in the aspect term, $W_j^a \in R^{d_H}$ is the semantic relevance weight of $H_j^a$ and $c_{pool}^{init}$, $W_2 \in R^{d_H \times d_H}$ and $W_3 \in R^{d_H \times d_H}$ are weights, and $b_2 \in R^{d_H}$ is the bias. These parameters are constantly updated during the learning process to achieve optimization.

Next, $W_j^a$ and $H_j^a$ are multiplied by each element to obtain the aspect term hidden vector $a_j$ which contains rich context semantic information. The calculation process is shown in Eq. (5):

$$a_j = W_j^a \odot H_j^a \tag{5}$$

where $a_j \in R^{d_H}$ is the hidden vector of the $j$th word in the aspect term obtained by the gating mechanism and $\odot$ represents the elementwise multiplication operation.

The pooling vector $a'_{pool}$ after the activation function is used as the final aspect term representation vector. Equation (6) shows the process of calculating the aspect pooling vector. To enhance the learning ability of $a_{pool}$, the *tanh*() activation function is used to adaptively adjust $a_{pool}$ to combine it more effectively with the context hidden vector $H^c$. The calculation process of $a'_{pool}$ is shown in Eq. (7):

$$a_{pool} = \sum_{j=1}^{m} a_j / m \tag{6}$$

$$a'_{pool} = Tanh\left(W_4 a_{pool} + b_3\right) \tag{7}$$

where $W_4 \in R^{d_H \times d_H}$ and $b_3 \in R^{d_H}$ are learnable parameters.

#### 3.4.3 Getting the context representation vector

Many interactive feature extraction structures directly use the average pooled vector of the aspect term to participate in the calculation of the context representation vector, but some aspect terms composed of multiple words may contain noise words. In addition, if the aspect term contains multiple target entities, they have different effects on sentiment classification. For this reason, this paper uses the aspect term representation vector $a'_{pool}$ which contains rich context relations for participation in the calculation of the context representation vector. Since $a'_{pool}$ undergoes gated calculation, it can reflect the most important word information in aspect terms while avoiding the interference of noise words.

*Sigmoid* gating is used to calculate the semantic correlation weight $W_j^c$ between the hidden vector of each context word and $a'_{pool}$. $W_j^c$ reflects the correlation degree of each context word and aspect term in each dimension. The calculation is shown in Eq. (8):

$$W_j^c = Sigmoid\left(W_5 H_j^c + W_6 a'_{pool} + b_4\right) \tag{8}$$

where $H_j^C \in R^{d_H}$ is the hidden vector of the $j$th context word, $W_j^c \in R^{d_H}$ is the weight of semantic relevance between $H_j$ and $a'_{pool}$, and $W_5 \in R^{d_H \times d_H}$, $W_6 \in R^{d_H \times d_H}$ and $b_2 \in R^{d_H}$ are learnable parameters.

Next, $W^c$ and $H^c$ are multiplied by each element to obtain the context representation vector $c$ based on the specific aspect term. The calculation process is shown in Eq. (9):

$$c_j = W_j^c \odot H_j^c \tag{9}$$

where $c_j \in R^{d_H}$ is the $j$th context word representation vector, and $\odot$ stands for the elementwise multiplication operation.

As shown in Eq. (10), the average pooling function is used to calculate the context pooling vector $c_{pool}$, and then the *Tanh*() activation function is used to enhance the learning ability of the context pooling vector to obtain the final context representation vector $c'_{pool}$. The calculation is shown in Eq. (11):

$$c_{pool} = \sum_{j=1}^{n} c_j / n \tag{10}$$

$$c'_{pool} = Tanh\left(W_7 c_{pool} + b_5\right) \tag{11}$$

where $W_4 \in R^{d_H \times d_H}$ and $b_5 \in R^{d_H}$ are learnable parameters.

### 3.5 Output layer

The final representation vector $O$ is composed of the context representation vector $c'_{pool}$ and the aspect term representation vector $a'_{pool}$. The combination vector $O$ is shown in Eq. (12):

$$O = merge\left(c'_{pool}, a'_{pool}\right) \tag{12}$$

where *merge*() represents combination methods. Experiments are carried out in two combinations of elementwise multiplication and concatenation.

As shown in Eqs. (13) and (14), the fully connected layer is used to map the combined vector $O$ into the classification space $C$, and finally the softmax layer is used to calculate the sentiment polarity.

$$x = W_O{}^T O + b_O \tag{13}$$

$$y = softmax(x) = \frac{\exp(x)}{\sum_{K=1}^{C} \exp(x)} \tag{14}$$

where $y \in R^C$ is the predicted emotional polarity distribution and $W_O \in R^{1 \times C}$ and $b_O \in R^C$ are learnable parameters.

### 3.6 Model training

#### 3.6.1 Objective function

As shown in Eq. (15), the CGBN model is trained by minimizing the cross-entropy loss function and the $L_2$ regular term:

$$L(\theta) = -\sum_{i=1}^{C} \hat{y} \log(y_i) + \lambda \|\theta\|^2 \tag{15}$$

where $\hat{y}$ represents the real sample, $y$ represents the predicted sentiment distribution, and $\lambda$ is the weight of the $L_2$ regular term.

#### 3.6.2 BERT-PT and domain adaptation of training sets

To alleviate the "domain bias" problem caused by the BERT language model in the pretraining phase, the BERT-PT pretraining file is fine-tuned with the restaurant and laptop datasets, which allows the model to learn domain-related features better. BERT-PT employs the corpus of Amazon laptop reviews (laptop domain) and Yelp Dataset Challenge reviews (restaurant domain). Furthermore, it has two pretraining tasks: MLM and NSP.

The MLM task is essential for injecting domain knowledge and reducing the "domain bias" of the BERT benchmark pretraining dataset. For example, in the sentence "The [*MASK*] is bright", the BERT using the Wikipedia corpus for the pretraining task may predict [*MASK*] as "sun", while the BERT-PT using the domain-related corpus for the pretraining task may predict [*MASK*] as "screen". NSP tasks can also allow the model to recognize domain-related features better. When the BERT training data are very limited, fine-tuning is not enough to ensure that the model fully understands the task. Therefore, task knowledge needs to be consolidated in the pretraining stage, and BERT-PT employs the SQuAD1.1 (large MRC dataset) dataset for pretraining tasks. MRC is a general task that can answer almost all questions about the content of documents; hence, a large MRC supervised corpus may also benefit the ATSA task [14].

The CGBN model also applies domain adaptation methods to the restaurant and laptop datasets during the training process. Domain adaptation is a representative method of transfer learning. The idea is to map data features of different domains (such as two different data sets) to the same feature space. Two advantages are as follows. First, data from other domains can be applied to enhance the training of the target domain, and second, the information-rich source domain samples can be used to improve the performance of the target domain model. The model cannot be fully trained due to the small training set of restaurant and laptop, and the CGBN model combines the training set of restaurant with laptop to address this problem. The model can adjust the parameters of the target field adaptively, allowing itself to be more fully trained. Moreover, BERT's ability to learn semantic features can be improved in the fine-tuning stage. The impact of various training sets on the target training set is presented in Section 4.4.2.

## 4 Experiments

### 4.1 Datasets and experimental settings

The CGBN model is evaluated on three public datasets. They are the Restaurant and Laptop datasets of SemEval-2014 Task4 and the ACL − 14 Twitter dataset, which are labeled as three emotional polarities: Positive, Neutral, and Negative. The specific statistics are shown in Table 1.

For our experiment, the batch size is set to 32, the epoch is set to 5, the dimension of the BERT hidden vector is set to 768, the optimizer is Adam, and the weights and deviations of the model are initialized with a Xavier uniform distribution. For laptop and Twitter fields, the learning rate is set to 2e-5, dropout is set to 0.1, and the $L_2$ regularization coefficient $\lambda$ is set to 0.01. For the restaurant domain, the learning rate is set to 3e-5, dropout is set to 0.025, and $\lambda$ is set to 2e-3. The BERT-Base (uncased) framework is the basis of the entire model, and the BERT-PT pretraining files enhance fine-tuning capabilities. To strengthen domain adaptation, the training sets of restaurants and laptops are combined. Then, their data domains are analyzed separately.

**Table 1** Statistics of datasets

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

### 4.2 Model comparisons

Classification accuracy and macro-F1 metrics are used as evaluation indicators to evaluate the performance of the CGBN model. To test the effectiveness of the model, this paper compares the CGBN model with many benchmark methods. A random initialization method is adopted to run each program 10 times and demonstrate the performance with "mean ± std". Table 2 shows the comparison results, and the best score for each column is shown in boldface. The introduction to the benchmark model is as follows:

**FDN** [17] proposes a double-layer gating mechanism to realize the interaction between aspect and context. It can also filter noise unrelated to aspect and highlight relevant features.

**MSAT** [18] designs a dynamic target representation module, which dynamically calculates the target representation vector based on each word in a sentence.

**IPAN** [19] regards POS information as the basis for calculating attention weights, strengthening the weight of words with specific parts of speech and finding more relevant opinion words.

**R-GAT** [20] reshapes and prunes the ordinary dependency tree and constructs a dependency tree with aspect as the root node to focus more on aspect words.

**RAO-CNN** [22] adopts the residual attention mechanism to control the flow of emotional information and encodes other words to reduce its interference in the feature fusion stage.

**IMAN** [24] uses an attention mechanism and convolution to encode context and aspect words. It adopts an attention mechanism to extract interactive features.

**MGAN** [25] uses multigrained attention to capture the interaction between aspect and context at the word level.

**AEN-BERT** [26] extracts the representation vectors of the context and aspect words by using a BERT encoder. It

**Table 2** The comparison of different models' performance

| Models | Restaurant | | Laptops | | Twitter | |
|---|---|---|---|---|---|---|
| | Accuracy | F1 | Accuracy | F1 | Accuracy | F1 |
| Embedding | | | | | | |
| FDN | 82.3 | 75.0 | 76.8 | 72.5 | 73.7 | 72.2 |
| MSAT | 81.43 | 72.63 | 78.21 | 74.31 | 74.63 | 73.22 |
| IPAN-LSTM | 82.8 | 73.8 | 77.2 | 73.5 | 74.3 | 72.5 |
| R-GAT | 83.30 | 76.08 | 77.42 | 73.76 | 75.57 | 73.82 |
| RAO-CNN | 80.98±0.52 | 71.34±0.65 | 74.45±0.73 | 69.72±1.18 | 73.12±0.95 | 71.20±1.42 |
| MGAN | 81.25 | 71.94 | 75.39 | 72.47 | 72.54 | 70.81 |
| BERT Models | | | | | | |
| MAST-BERT | 83.39 | 76.10 | 80.25 | 76.79 | 75.87 | 74.36 |
| IPAN-BERT | 85.9 | 76.4 | 78.5 | 76.0 | 76.7 | 75.9 |
| R-GAT-BERT | 86.60 | 81.35 | 78.21 | 74.07 | 76.15 | 74.88 |
| IMAN | 83.95 | 75.63 | 80.53 | 76.91 | 75.72 | 74.50 |
| AEN-BERT | 82.50±0.45 | 72.72±0.87 | 79.07±0.87 | 75.54±0.81 | 74.20±0.51 | 73.00±0.68 |
| TD-BERT* | 85.10±0.20 | 78.35±1.34 | 78.87±1.13 | 74.38±0.81 | 76.69±0.58 | 74.28±0.68 |
| BERT-ADA | 87.89 | 81.05 | 80.23 | 75.77 | – | – |
| LCF-BERT | 87.14 | 81.74 | 82.45 | 79.59 | 77.31 | 75.78 |
| Ours | | | | | | |
| CGBN | 86.47±0.85 | 80.34±1.09 | 80.33±0.86 | 76.57±1.32 | **77.53±0.65** | **76.16±0.55** |
| CGBN-PT-DA | **89.15±0.49** | **84.40±0.80** | **82.76±0.63** | **79.62±0.87** | – | **–** |

(1) "*" indicates that the model has multiple architectures, with the highest performance in the Table. (2) "-" means unreported experimental results

also adopts a mechanism to extract features of the relationship between the vectors.

**TD-BERT** [37] proposed a new method of modeling aspect words. It explores the performance of extracting aspect word vectors from context hidden vectors.

**BERT-ADA** [35] adopts domain-related knowledge to pretrain BERT and applies domain adaptation to ATSA tasks. The researchers propose three combinations of training sets: in-domain training sets, cross-domain training sets, and joint-domain training sets. These different combinations enhance the domain adaptability of the model.

**LCF-BERT** [38] focuses on the local context information for specific aspect words by applying the context features dynamic mask (CDM) and context features dynamic weighted (CDW) layers. It also uses MHSA to capture both local context and global context information.

Table 2 compares the results from each model. The performances of models using traditional GloVe or Word2Vec embedding vectors are generally inferior to those of BERT-based models. The performances, which construct complex feature extraction modules, are constrained by the shortcomings of traditional word embedding vectors. When the three models, IPAN, R-GAT, and MAST, use BERT to calculate the embedding vector, the performance is significantly improved.

AEN-BERT, MAST-BERT, IPAN-BERT, and IMAN are models based on the BERT encoder. Although their feature extraction layers are carefully designed to capture the semantic relationship between context and aspect, their average performances are worse than or equivalent to that of TD-BERT. In the TD-BERT model, the aspect vector extracted from the context hidden vector contains rich contextual semantic-related information that can avoid the interference of irrelevant words. This reflects the advantage of extracting aspect word embedding information from context hidden vectors.

R-GAT-BERT obtains its embedding vector from BERT. It obtains dependencies directly related to specific aspect terms by means of an aspect-oriented dependency tree. Therefore, it has the greatest improvement using the restaurant dataset. However, there are still shortcomings in feature interaction due to the employment of the traditional BERT encoding method. The BERT-ADA model applies domain knowledge and a training set fusion method to enhance the feature learning ability in the pretraining and fine-tuning stages of the BERT encoder, so it achieves a good result. However, it only employs $H_{[CLS]}$ as the final representation vector without considering the semantic information of context and aspect. LCF-BERT captures both local and global

context information, so it has strong performance but does not consider the use of domain-related knowledge.

The CGBN model extracts both context and aspect hidden vectors simultaneously in the last hidden layer of the BERT encoder. It employs the cogate network to further extract the interactive features of context and aspect terms. It also improves the domain adaptability of the BERT pretraining and fine-tuning stages using the restaurant and laptop datasets, so it achieves the best results.

## 4.3 Exploring different ways to obtain embedding vectors

To explore the impact of different forms of embedding vector acquisition methods on model performance, experiments were conducted on four models. Table 3 shows the experimental results. The performance of models based on two BERT encoders is much worse than that of models based on one BERT encoder. This is because the correlation between the context and aspect hidden vectors extracted by two independent BERT encoders is poor. When dealing with complex sentences containing multiple aspect words, these hidden vectors have difficulty judging the emotional polarity of a specific aspect word. For models that only use context as the input of the BERT encoder, the extracted context and aspect hidden vectors have rich semantic correlation with each other, so a model that uses the BERT single sentence encoder can have greatly improved performance.

## 4.4 Analyze the CGBN model

### 4.4.1 Ablation experiment

To analyze the importance of each module of the CGBN, an ablation experiment was designed. The experimental variables are shown below, and the experimental results are shown in Table 4.

**w/o gate-a**: Gating is not used for feature extraction of the aspect term, and the aspect average pooling vector after *Tanh*() activation function is directly used as its representation vector.

**w/o gate-c**: Gating is not used when extracting context features, and the pooling vector $c_{pool}^{init}$ is used as the context representation vector.

**w/o gate**: Instead of using co-gate network feature extraction, the average pooling vector of aspect and context after *tanh*() activation function is taken as the final representation vector.

**w/o aspect**: The final representation vector $O$ does not combine the aspect pooling vector $a_{pool}^{'}$.

**w/o context**: The final representation vector $O$ does not combine the context pooling vector $c_{pool}^{'}$.

**w/o $H_{[CLS]}$**:The final representation vector does not combine $H_{[CLS]}$ vector.

**CGBN-PT**: Use BERT-PT pre-training files in restaurant and laptop experiments

**CGBN-DA**: Use the combined training set of restaurant and laptop, and test these two target areas.

Table 4 shows that in terms of accuracy and macro-F1, the performance of each CGBN ablation model is inferior to that of the CGBN model.

Compared with the CGBN model, the performance of the 'w/o gate-c' experiment is poor, which indicates that using the average pooling vector of the aspect term to model the context will introduce noise words. It also shows that the cogate mechanism can more effectively capture the relationship between context and the aspect term than the traditional interactive gating mechanism. According to the results of the 'w/o gate-c' experiment, it is necessary to use the gating mechanism to calculate the context representation vector based on a specific aspect term because the gating

**Table 3** We explored the impact on performance of different ways of using the BERT encoder to obtain the context and aspect hidden vectors on the four models

| Models | Restaurant | | Laptops | | Twitter | |
|---|---|---|---|---|---|---|
| | acc | f1 | acc | f1 | acc | f1 |
| Use two independent BERT encoders | | | | | | |
| FDN-double-BERT | 83.35 ± 0.85 | 75.56 ± 1.75 | 78.45 ± 1.18 | 74.54 ± 1.59 | 73.48 ± 0.65 | 71.95 ± 0.53 |
| MGAN-double-BERT | 84.06 ± 0.45 | 77.69 ± 0.81 | 79.23 ± 0.39 | 75.61 ± 0.54 | 75.25 ± 0.45 | 73.86 ± 0.50 |
| AEN-double-BERT | 82.50 ± 0.45 | 72.72 ± 0.87 | 79.07 ± 0.87 | 75.54 ± 0.81 | 74.20 ± 0.51 | 73.00 ± 0.68 |
| CGBN-double-BERT | 83.66 ± 0.98 | 75.69 ± 1.38 | 79.55 ± 0.55 | 75.26 ± 1.01 | 74.57 ± 0.73 | 73.49 ± 0.89 |
| Use a single BERT encoder | | | | | | |
| FDN-single-BERT | 86.21 ± 0.50 | 80.19 ± 0.70 | 80.10 ± 0.79 | **76.98 ± 0.89** | 77.24 ± 0.65 | 75.76 ± 0.60 |
| MGAN-single-BERT | 85.89 ± 0.81 | 80.04 ± 1.14 | 79.62 ± 0.94 | 75.70 ± 0.72 | 76.58 ± 0.65 | 74.95 ± 0.42 |
| AEN-single-BERT | 85.09 ± 0.54 | 78.48 ± 1.24 | 79.78 ± 0.63 | 75.90 ± 0.52 | 75.65 ± 0.51 | 74.15 ± 0.60 |
| CGBN | **86.47 ± 0.85** | **80.34 ± 1.09** | **80.33 ± 0.86** | 76.57 ± 1.32 | **77.53 ± 0.65** | **76.16 ± 0.55** |

These two methods are shown in Figs. 2 and 3 respectively

**Table 4** Ablation experiment of CGBN

| Models | Restaurant | | Laptops | | Twitter | |
|---|---|---|---|---|---|---|
| | Accuracy | F1 | Accuracy | F1 | Accuracy | F1 |
| CGBN ablations | | | | | | |
| w/o gate-a | 86.03 ± 0.32 | 79.55 ± 0.79 | 79.86 ± 0.69 | 75.98 ± 0.86 | 77.25± 0.58 | 75.84± 0.67 |
| w/o gate-c | 86.16 ± 0.98 | 80.05 ± 0.99 | 79.39 ± 0.71 | 76.08 ± 0.87 | 77.13± 0.44 | 75.89± 0.74 |
| w/o gate | 85.94 ± 0.49 | 79.94 ± 1.04 | 79.25 ± 0.55 | 75.56 ± 0.64 | 77.07± 0.58 | 75.89± 1.00 |
| w/o aspect | 85.40 ± 0.67 | 79.85 ± 1.41 | 79.39 ± 0.71 | 75.37 ± 1.07 | 75.07± 0.94 | 73.88± 1.31 |
| w/o context | 85.54 ± 0.36 | 79.52 ± 0.64 | 79.08 ± 0.71 | 75.41 ± 1.01 | 76.88± 0.29 | 75.60± 0.37 |
| CGBN-PT | 88.53 ± 0.58 | 83.27 ± 0.90 | 81.03 ± 0.94 | 77.52 ± 1.25 | – | – |
| CGBN-DA | 87.50 ± 0.71 | 82.02 ± 1.36 | 81.11 ± 0.86 | 77.98 ± 0.95 | – | – |
| Ours | | | | | | |
| CGBN | 86.47 ± 0.85 | 80.34 ± 1.09 | 80.33 ± 0.86 | 76.57 ± 1.32 | **77.53± 0.65** | **76.16± 0.55** |
| CGBN-PT-DA | **89.15 ± 0.49** | **84.40 ± 0.80** | **82.76 ± 0.63** | **79.62 ± 0.87** | – | – |

mechanism determines the degree of relevance between the given aspect term and the context representation vector. Therefore, the mechanism can effectively avoid the influence of irrelevant words in the context of the results. The effect of the 'w/o gate' experiment is inferior to that of 'w/o gate-a' and 'w/o gate-c' because this 'w/o gate' experiment adopts only the pooling vector of context and aspect terms as its representation vector without further extracting interactive features.

The representation vectors of context and aspect calculated by the co-gate mechanism contain rich global semantic information. The CGBN model takes the combined vector of context and aspect as the final representation vector. According to the experiments 'w/o aspect' and 'w/o context', the performance of the model in either case is worse than the performance of CGBN with both attributes, which shows that both are effective.

The CGBN model applies the pretraining parameters of BERT-PT to fine-tuning using the restaurant and laptop datasets. Based on the results, we can see that the BERT-PT pretraining parameters improve the study model domain-related features and greatly improve the model performance. To explore the domain adaptation of the model, the training sets of restaurants and laptops are combined and tested in those two domains. The performance of the model is significantly improved, which indicates that the CGBN model can improve the performance of the target domain by using source domain samples with rich information. The model cannot be fully trained because of the limited original training; however, the fused training set can fill the gap. When the BERT-PT pretraining files and fusion training sets are used simultaneously, the model achieves the best performance in these two areas, which shows that it is very effective in using domain knowledge to enhance the learning of related domain knowledge. A more detailed description is in Section 4.4.2.

#### 4.4.2 Domain adaptation of training set

To explore the domain adaptability of the CGBN model and test the impact of different training sets on performance, the model was tested with multiple training sets. As shown in Table 5, 'rest.', 'lap.' and 'twi.' represent the datasets restaurants, laptops and twitter, respectively. 'rest. + lap.','rest. + twi.','lap. + twi.' and 'rest. + lap. + twi.' represent four different combined training sets. According to the different test sets, the three single training sets can be divided into in-domain and cross-domain types, and the four combined training sets are of the joint-domain type [36].

The effect of a cross-domain training set is worse than that of an in-domain training set, which may be due to the difference of domain knowledge between different domain data sets; this weakens the feature learning ability of the model for the target domain. Specifically, the restaurant domain is about food, service, atmosphere, price, quality, etc., while the laptop domain is about quality, price, memory, weight, storage space, CPU, etc., and the twitter domain contains even more complex domain knowledge, such as entertainment stars, politicians, news and current events. There are different degrees of knowledge deviation between fields, so their impact on performance is also varied.

For the restaurant and laptop target areas, the joint-domain training set combined with 'twi' seems to have an adverse effect on performance. This may be a large difference in the

**Table 5** Test the CGBN model on 7 training sets

| Training set | Restaurant | | | Laptops | | | Twitter | | |
|---|---|---|---|---|---|---|---|---|---|
| | Train type | Accuracy | F1 | Train type | Accuracy | F1 | Train type | Accuracy | F1 |
| rest. | In | 86.47 ± 0.85 | 80.34 ± 1.09 | Cross | 79.47 ± 0.79 | 76.04 ± 0.93 | Cross | 56.29 ± 2.53 | 56.08 ± 2.14 |
| lap. | Cross | 82.19 ± 1.12 | 74.20 ± 1.07 | In | 80.33 ± 0.86 | 76.57 ± 1.32 | Cross | 58.24 ± 1.45 | 58.26 ± 1.25 |
| twi. | Cross | 75.89 ± 1.34 | 68.21 ± 1.73 | Cross | 73.67 ± 1.41 | 68.56 ± 1.58 | In | **77.53 ± 0.65** | **76.16 ± 0.55** |
| rest.+lap. | joint | **87.50 ± 0.71** | **82.02 ± 1.36** | joint | **81.11 ± 0.86** | **77.98 ± 0.95** | joint | 59.33 ± 0.79 | 59.44 ± 0.64 |
| rest.+twi. | joint | 85.76 ± 0.58 | 80.09 ± 0.56 | joint | 78.68 ± 0.47 | 75.04 ± 0.99 | joint | 76.85 ± 0.65 | 75.45 ± 1.14 |
| lap.+twi. | joint | 80.94 ± 0.49 | 74.49 ± 1.53 | joint | 79.70 ± 0.55 | 75.86 ± 0.69 | joint | 76.59 ± 0.58 | 74.88 ± 0.97 |
| rest.+lap.+twi. | joint | 86.26 ± 0.63 | 80.04 ± 0.88 | joint | 80.49 ± 0.40 | 76.98 ± 0.48 | joint | 76.59 ± 0.72 | 75.36 ± 0.52 |

label distribution and data distribution between the twitter data set and the restaurant and laptop data sets. Twitter's training sets of neutral label samples accounted for 50% of their total, much higher than in the other two data sets, and neutral emotion is a vague emotional state; a high proportion of neutral training samples will increase the difficulty of model training [26]. Regarding the difference in data distribution, on the one hand, the twitter dataset contains many complex sentences (such as satirical sentences), and its grammatical correctness is low. On the other hand, the domain knowledge of twitter is quite different than the domain knowledge of Restaurant and Laptop.

In addition, the performance from a joint-domain training set that includes the same field as the target is better than that of one that does not. For example, when testing the restaurant target field, the performance from using the 'rest. + lap.', 'rest. + twi.' and 'rest. + lap. + twi.' training sets is better than it is from using 'lap. + twi.'.

To improve the training effect of a joint domain, the combined training set should contain the target domain training set, which allows the model to obtain enough knowledge in its field of interest. Furthermore, the differences of domain and label distribution in these combined training sets should not be too large, so that the model can be trained more fully, and the semantic features of BERT in the fine-tuning stage can have improved learning ability.

#### 4.4.3 Pooling method and connection strategy

This section will explore the impact of the pooling method and the connection method of the final representation vector on performance to confirm the effectiveness of the CGBN model. The experimental results are summarized in Table 6. The results show that the model with 'Con' connection strategy and 'avg' pooling method has the best performance.

#### 4.4.4 Case study

We select example sentences from the restaurant and laptop data sets to further study the impact of the co-gate mechanism and the BERT hidden vector extraction method on performance. For Table 7, CGBN-W/O-gate removes the co-gate mechanism but directly connects the average pooling vector of context and aspect words. CGBN-double-BERT uses two independent BERT encoders to extract features of context and aspect words.

From sentences 1 to 4, only the CGBN and CGBN W/O-gate can correctly judge the emotional polarity of all aspect words, which reflects the advantage of extracting context and aspect hidden vectors from a BERT encoder at the same time. CGBN-double-BERT is easily disturbed by opinion words with different emotions. It is difficult to accurately find key information when dealing with complex sentences.

**Table 6** Experiments on pooling methods and connection strategies

| Combination strategy | Pooling method | Restaurant | | Laptops | | Twitter | |
|---|---|---|---|---|---|---|---|
| | | Accuracy | F1 | Accuracy | F1 | Accuracy | F1 |
| Mul | max | 85.94 ± 0.67 | 79.88 ± 0.95 | 79.47 ± 0.79 | 75.84 ± 1.13 | 77.32 ± 0.44 | 75.79 ± 0.55 |
| | avg | 85.90 ± 0.81 | 79.94 ± 1.28 | 79.47 ± 0.47 | 76.23 ± 1.17 | 76.81 ± 0.80 | 75.11 ± 0.81 |
| Con | max | 86.11 ± 0.51 | 79.73 ± 1.21 | 79.94 ± 0.94 | 76.50 ± 1.24 | 77.25 ± 0.51 | 75.73 ± 0.61 |
| | avg | **86.47 ± 0.85** | **80.34 ± 1.09** | **80.33 ± 0.86** | **76.57 ± 1.32** | **77.53 ± 0.65** | **76.16 ± 0.55** |

In the combination strategy, 'Mul' means element-wise multiplication, and 'Con' means concatenation

**Table 7** Case study

| No. | sentence | CGBN | CGBN W/O-gate | CGBN-double-BERT |
|---|---|---|---|---|
| 1 | Great [**food**]$_1$ but the [**service**]$_{-1}$ is dreadful. | 1, −1 | 1, −1 | 1, −1 |
| 2 | The [**food**]$_1$ is so good and so popular that [**waiting**]$_{-1}$ can really be a nightmare. | 1, −1 | 1, −1 | 1, 1 $_x$ |
| 3 | After replacing the [**hard drive**]$_0$, the [**battery**]$_{-1}$ stopped working, which was frustrating. | 0, -1 | 0, -1 | -1 $_x$, -1 |
| 4 | [**Drivers**]$_1$ updated ok but the [**BIOS update**]$_{-1}$ froze the [**system**]$_{-1}$ up and the computer shut down . | 1, -1, -1 | 1, -1, -1 | -1 $_x$, -1, -1 |
| 5 | In mi burrito, here was nothing but dark [**chicken**]$_{-1}$ that had that cooked last week and just warmed up in a microwave [**taste**]$_{-1}$. | -1, -1 | -1, 0 $_x$ | -1, -1 |
| 6 | The [**staff**]$_{-1}$ should be a bit more friendly. | -1 | 1 $_x$ | 1 $_x$ |
| 7 | They did not have [**mayonnaise**]$_{-1}$, forgot our [**toast**]$_{-1}$, left out [**ingredients**]$_{-1}$ -LRB- ie [**cheese**]$_0$ in an [**omelet**]$_0$ -RRB-, below hot temperatures and the [**bacon**]$_{-1}$ was so over cooked it crumbled on the [**plate**]$_0$ when you touched it. | -1, -1, -1, 0, 0, -1, 0 | -1, -1, -1, -1 $_x$, 0, -1, -1 $_x$ | -1, -1, -1, -1 $_x$, −1 $_x$, -1, -1 $_x$ |
| 8 | The [**pizza**]$_1$ is the best if you like [**thin crusted pizza**]$_0$. | 1, 0 | 1, 1 $_x$ | 1, 1 $_x$ |

(1) The words in bold and square brackets in the sentence are aspect words, and the subscripts 1, 0, −1 represent emotional tags, which correspond to positive, neutral and negative emotional polarities, respectively. (2) The results predicted by the models are displayed in the order of appearance of the aspect words in the sentence. (3) "x" means that the emotion prediction is wrong

In sentence 5, for the aspect word "taste", only CGBN W/O-gate model makes a wrong judgment. In this model, the pooling vector of context and aspect word is taken as the final representation vector. Sigmoid gating is not used to further compute deep semantic features. Therefore, the influence of more important words in context is ignored.

From Sentence 6 to Sentence 8, only the CGBN model makes correct judgments for all examples. Sentence 7 includes seven aspects with negative or neutral polarity. This sentence contains more negative opinion words, so CGBN-double-BERT judges all polarities as negative; CGBN W/O-gate also judges the polarity of the two neutral emotion aspect words as negative.

The above example sentences prove that the co-gate mechanism and the use of a BERT encoder for feature extraction are effective. The CGBN model can make accurate predictions when facing some complex sentences.

## 5 Conclusion

CGBN models based on a single BERT encoder, co-gate mechanism, and domain adaptation are proposed in this paper. Unlike general models that use two BERT encoders to extract context and aspect vectors, the CGBN model only adopts sentence sequences as the input to the BERT encoder to obtain both context and aspect hidden vectors. This not only reduces the computational load but also allows the initial vector to contain rich semantic interaction information, which is helpful for the final emotion prediction. A new type of nonparallel interactive gating mechanism called a co-gate is also proposed. This mechanism first fuses context information into the aspect vector to highlight the influence of important words on the aspect word. After that, it calculates the context representation based on the aspect representation vector. The deep semantics of the context can be explored, thus reducing unnecessary interference from the context. In addition, the CGBN model explores the impact of different combinations of training sets and BERT-PT on model performance. Experiments show that our CGBN model is always superior to the latest technology on SemEval2014 and twitter datasets.

In our future research, we will experiment with the model in a multilingual environment. We will also explore the influence of different pretraining models on performance.

**Acknowledgements** This work was supported in part by the Post-graduate's Innovation Fund Project of Hebei Province (No. CXZZSS2021043) and in part by Natural Science Foundation of Hebei Province (No. F2021202038).

## References

1. Rana TA, Cheah Y-N (2016) Aspect extraction in sentiment analysis: comparative analysis and survey. Artif Intell Rev 46:459–483. https://doi.org/10.1007/s10462-016-9472-z
2. Appel O, Chiclana F, Carter J, Fujita H (2017) A consensus approach to the sentiment analysis problem driven by support-based IOWA majority. Int J Intell Syst 32:947–965. https://doi.org/10.1002/int.21878
3. Appel O, Chiclana F, Carter J, Fujita H (2017) Cross-ratio uninorms as an effective aggregation mechanism in sentiment analysis. Knowl Based Syst 124:16–22. https://doi.org/10.1016/j.knosys.2017.02.028
4. Dosoula N, Griep R, Den Ridder R et al (2016) Sentiment analysis of multiple implicit features per sentence in consumer review data.

Front Artif Intell Appl:241–254. https://doi.org/10.3233/978-1-61499-714-6-241
5. Do HH, Prasad P, Maag A, Alsadoon A (2019) Deep learning for aspect-based sentiment analysis: a comparative review. Expert Syst Appl 118:272–299. https://doi.org/10.1016/j.eswa.2018.10.003
6. Xue W, Li T (2018) Aspect based sentiment analysis with gated convolutional networks. In: Proceedings of the 56th annual meeting of the association for computational linguistics, vol 2018, pp 2514–2523
7. Pennington J, Socher R, Manning C (2014) Glove: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1532–1543
8. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv:1301.3781
9. J Devlin, M-W Chang, K Lee, K Toutanova (2019) BERT: Pre-training of deep bidirectional transformers for language Understanding, Proc. NAACL-HLT, pp 4171–4186 2019
10. Vaswani A, Brain G, Shazeer N et al (2017) Attention is all you need. Adv Neural Inf Process Syst 30:5998–6008
11. Su J, Yu S, Luo D (2020) Enhancing aspect-based sentiment analysis with capsule network. IEEE Access 8:100551–100561. https://doi.org/10.1109/ACCESS.2020.2997675
12. Yang C, Zhang H, Jiang B, Li K (2019) Aspect-based sentiment analysis with alternating coattention networks. Inf Process Manag 56:463–478. https://doi.org/10.1016/j.ipm.2018.12.004
13. He R, Lee WS, Ng HT, Dahlmeier D (2018) Exploiting document knowledge for aspect-level sentiment classification. In: Proceedings of the 56th annual meeting of the Association for Computational Linguistics (volume 2: short papers). Association for Computational Linguistics, Stroudsburg, PA, USA, pp 579–585
14. Hu X, Bing L, Lei S, Philip YS (2019) BERT post-training for review reading comprehension and aspect-based sentiment analysis. Proc NAACL:2324–2335
15. Dong L, Wei F, Tan C, Tang D, Zhou M, Xu K (2014) Adaptive recursive neural network for target-dependent twitter sentiment classification. In: Proceedings of the 52nd annual meeting of the association for computational linguistics, vol 2: short papers, pp 49–54
16. Park H, Song M, Shin K-S (2020) Deep learning models and datasets for aspect term sentiment classification: implementing holistic recurrent attention on target-dependent memories. Knowl Based Syst 187:104825. https://doi.org/10.1016/j.knosys.2019.06.033
17. Shuang K, Yang Q, Loo J, Li R, Gu M (2020) Feature distillation network for aspect-based sentiment analysis. Inf Fusion 61:13–23. https://doi.org/10.1016/j.inffus.2020.03.003
18. Lin Y, Wang C, Song H, Li Y (2021) Multi-head self-attention transformation networks for aspect-based sentiment analysis. IEEE Access 9:8762–8770. https://doi.org/10.1109/ACCESS.2021.3049294
19. Wang K, Shen W, Yang Y, et al (2020) Relational graph attention network for aspect-based sentiment analysis. In: Proceedings of the 58th annual meeting of the association for computational linguistics, pp 3229–3238
20. Shuang K, Gu M, Li R, Loo J, Su S (2021) Interactive POS-aware network for aspect-level sentiment classification. Neurocomputing 420:181–196. https://doi.org/10.1016/j.neucom.2020.08.013
21. Lv Y, Wei F, Cao L et al (2021) Aspect-level sentiment analysis using context and aspect memory network. Neurocomputing 428:195–205. https://doi.org/10.1016/j.neucom.2020.11.049
22. Wu C, Xiong Q, Yang Z, Gao M, Li Q, Yu Y, Wang K, Zhu Q (2021) Residual attention and other aspects module for aspect-based sentiment analysis. Neurocomputing. 435:42–52. https://doi.org/10.1016/j.neucom.2021.01.019
23. Liu MZ, Zhou FY, Chen K, Zhao Y (2021) Co-attention networks based on aspect and context for aspect-level sentiment analysis. Knowl Based Syst 217:106810. https://doi.org/10.1016/j.knosys.2021.106810
24. Zhang Q, Lu R, Wang Q, Zhu Z, Liu P (2019) Interactive multi-head attention networks for aspect-level sentiment classification. IEEE Access 7:160017–160028. https://doi.org/10.1109/ACCESS.2019.2951283
25. Fan F, Feng Y, Zhao D (2018) Multi-grained Attention Network for Aspect-Level Sentiment Classification. In: Proceedings of the 2018 Conference on empirical methods in natural language processing. Association for Computational Linguistics, pp 3433–3442
26. Song Y, Wang J, Jiang T et al (2019) Targeted sentiment classification with attentional encoder network. In: Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics), pp 93–103
27. Li X, Fu X, Xu G, Yang Y, Wang J, Jin L, Liu Q, Xiang T (2020) Enhancing BERT representation with context-aware embedding for aspect-based sentiment analysis. IEEE Access 8:46868–46876. https://doi.org/10.1109/ACCESS.2020.2978511
28. Kumar A, Narapareddy VT, Aditya Srikanth V, Neti LBM, Malapati A (2020) Aspect-based sentiment classification using interactive gated convolutional network. IEEE Access 8:22445–22453. https://doi.org/10.1109/ACCESS.2020.2970030
29. Sun K, Zhang R, Mensah S et al (2019) Aspect-level sentiment analysis via convolution over dependency tree. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), pp 5678–5687
30. Zhang C, Li Q, Song D (2019) Aspect-based sentiment classification with aspect-specific graph convolutional networks. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), pp 4567–4577
31. Chen F, Huang Y (2019) Knowledge-enhanced neural networks for sentiment analysis of Chinese reviews. Neurocomputing. 368:51–58. https://doi.org/10.1016/j.neucom.2019.08.054
32. Peters M, Neumann M, Iyyer M et al (2018) Deep contextualized word representations. In: Proceedings of the 2018 conference of the north American chapter of the Association for Computational Linguistics: human language technologies, volume 1 (long papers), pp 2227–2237
33. Peters ME, Neumann M, Iyyer M et al (2018) Improving language understanding by generative pre-training. OpenAI
34. Yang Z, Dai Z, Yang Y et al (2019) XLNet: generalized autoregressive pretraining for language understanding. In: Advances in Neural Information Processing Systems, pp 5754–5764
35. Rietzler A, Stabinger S, Opitz P, Engl S (2019) Adapt or get left behind: domain adaptation through BERT language model Finetuning for aspect-target sentiment classification. Proc 12th Conf Lang Resour Eval (LREC 2020) 4933–4941
36. Rana TA, Cheah Y-N (2017) A two-fold rule-based model for aspect extraction. Expert Syst Appl 89:273–285. https://doi.org/10.1016/j.eswa.2017.07.047
37. Gao Z, Feng A, Song X, Wu X (2019) Target-dependent sentiment classification with BERT. IEEE Access 7:154290–154299. https://doi.org/10.1109/ACCESS.2019.2946594
38. Zeng B, Yang H, Xu R, Zhou W, Han X (2019) LCF: a local context focus mechanism for aspect-based sentiment classification. Appl Sci 9:3389. https://doi.org/10.3390/app9163389