Feng *et al. Journal of Big Data* (2023) 10:176
https://doi.org/10.1186/s40537-023-00856-8

Journal of Big Data

RESEARCH Open Access

Aspect-level sentiment classification with fused local and global context

Ao Feng[1], Jiazhi Cai[1*], Zhengjie Gao[1] and Xiaojie Li[1]

*Correspondence:
cjzlogirl@qq.com

[1] Chengdu University of Information Technology, Chengdu, China

Abstract

Sentiment analysis aims to determine the sentiment orientation of a text piece (sentence or document), but many practical applications require more in-depth analysis, which makes finer-grained sentiment classification the ideal solution. Aspect-level Sentiment Classification (ALSC) is a task that identifies the emotional polarity for aspect terms in a sentence. As the mainstream Transformer framework in sentiment classification, BERT-based models apply self-attention mechanism that extracts global semantic information for a given aspect, while a certain proportion of local information is missing in the process. Although recent ALSC models have achieved good performance, they suffer from robustness issues. In addition, uneven distribution of samples greatly hurts model performance. To address these issues, we present the PConvBERT (Prompt-ConvBERT) and PConvRoBERTa (Prompt-ConvRoBERTa) models, in which local context features learned by a Local Semantic Feature Extractor (LSFE) are fused with the BERT/RoBERTa global features. To deal with the robustness problem of many deep learning models, adversarial training is applied to increase model stability. Additionally, Focal Loss is applied to alleviate the impact of unbalanced sample distribution. To fully explore the ability of the pre-training model itself, we also propose natural language prompt approaches that better solve the ALSC problem. We utilize masked vector outputs of templates for sentiment classification. Extensive experiments on public datasets demonstrate the effectiveness of our model.

Keywords: Aspect-level sentiment classification, Local semantic feature extractor, Adversarial training

Introduction

With the rapid development of economy and society, the Internet is growing with a very fast pace. At this time, the Internet contains a huge amount of information filled with rich text and other media. People are surrounded by all kinds of data every day. Review text on e-commerce platforms, live broadcast and movie websites can reflect users' interest towards their products, services or movies. Valuable information can be mined from these resources to adjust the products and services according to the comment user's preferences, and the goal is to meet the interest of related businesses. Sentiment analysis [1] is to mine users' opinions on products and services through text, and to obtain if and how much they like or hate the object. Sentiment analysis is an important Customer Relationship Management (CRM) tool for businesses.

Sentiment analysis can be divided into three categories according to the granularity of the research object: document-level, sentence-level and aspect-level. The first two belong to coarse-grained sentiment analysis, and Aspect-Level Sentiment Classification (ALSC) [2] is a subtask of fine-grained sentiment analysis [3, 4]. Coarse-grained sentiment analysis, which has been extensively studied and become an almost solved problem, focuses on the overall sentiment polarity of the whole text piece. It can't extract people's views on a specific object, especially when there are multiple objects involved in a sentence.

Existing works for ALSC have achieved promising results. Luo et al. [5] focused on modeling the interactions among aspect terms. They suggested that the previous approaches ignored the interaction between aspect terms and the label imbalance in the sequence labeling task, and proposed a GRACE framework to solve the end-to-end polar extraction problem. Ma et al. [6] used position bias for sentiment classification to improve the robustness of the model. They find that state-of-the-art ALSC models suffer from robustness problems, especially in two situations: (1) out-of-domain scenario; and (2) adversarial scenario. In order to solve this problem, a simple and effective induction bias was proposed, namely position bias. They proposed two mechanisms to capture position deviation, namely position-biased weight and position-biased dropout. Oh et al. [7] proposed deep context relation-aware networks for ALSC. It allowed interaction between subtasks with deep contextual information based on two modules, namely aspect and opinion propagation and explicit self-policing policies. In particular, they designed a novel self-policing strategy for ABSA (Aspect-Based Sentiment Analysis) with advantages in addressing multiple aspects. Chen et al. [8] believed that how to locate the corresponding opinion context for each aspect word was a key challenge in the task of ABSA. Recent studies hope to capture the interaction between aspect and opinion context through syntactic dependencies in dependency trees. While syntactic dependence can achieve certain improvements, it still faces some limitations. Specifically, they present a model for automatically inducing discrete opinion trees for each aspect, but these methods [5–8] don't use prompt learning, and local semantic information is often missing. In recent years, prompt learning [9, 10] for effective extraction of semantic information has become another promising approach in ALSC.

In this paper, we propose PConvBERT (Prompt-ConvBERT) model and PConvRoBERTa (Prompt-ConvRoBERTa) model based on the BERT/RoBERTa backbone. Features learned by Local Semantic Feature Extractor (LSFE) are fused with the global features from BERT/RoBERTa to get more comprehensive semantic representation. We also use prompt learning to create templates for input sentences, so that the upstream pre-training model can better complete the downstream task with less annotating data. Experimental results show that our approach achieves comparable performances to current state-of-the-art models.

Main contributions of this article can be summarized as follows:

1. We fuse the local context features learned by an CNN-based LSFE with the global context features learned by BERT/RoBERTa, which improves performance over each individual model.

2. Text template with cloze-type question prompts is proposed to fully exploit the ability of the pre-training model. We extract the feature vectors from the [Mask] position in the sentence for sentiment classification.
3. Experimental results show that our model achieves competitive performances. To better understand the performance improvement, we perform ablation experiments with important parameters and also show the performance impact of different prompt templates.

Related work

Traditional machine learning and deep learning methods

Traditional approaches of sentiment analysis can be roughly divided into lexicon-based approaches and machine learning models [11]. The accuracy of the former greatly depends on the quality of dictionaries. With of the huge manpower required and difficulty to generalize, this type of methods is being ignored by the community. Sentiment analysis based on machine learning mainly uses supervised learning methods. The commonly used classification methods include Support Vector Machine (SVM) and Naive Bayes (NB). The main difficulties are the design of classifiers, the acquisition of data and the interpretation of unseen phrases. Sentiment analysis based on machine learning needs to construct features manually, with data and features composition greatly affects accuracy. Moreover, fine-grained sentiment analysis needs to consider the target object in the sentence and its surrounding context, which is difficult to model for traditional machine learning methods.

With the development of deep learning, natural language processing tasks based on neural networks have attracted more attention. Huang et al. [12] added aspect information to the CNN model for sentiment classification, and the supervised model captured sentiment information of different aspects. Because CNN is difficult to capture long-distance features, Tang et al. [13] proposed Target-Dependent LSTM (TD-LSTM) and Target-Connected LSTM (TC-LSTM) methods based on Long Short-Term Memory (LSTM). The proposed method extended LSTM by considering aspect information. They considered given targets as features and related them to contextual features for aspect-level sentiment classification. Although LSTM has the ability to model aspects and context relationships, its sequential structure makes it difficult to parallelize. As the attention mechanism is gradually applied to natural language processing tasks, Wang et al. [14] proposed a target embedding LSTM method based on the attention mechanism, which proved to be an effective method to force the neural model to process the relevant parts of the sentence. The attention mechanism forced the model to focus on important parts of the sentence, so that the model has the ability to respond to the sentiment of specific aspects. Considering the temporal nature of sentences, Yang et al. [15] also proposed two bidirectional LSTMs based on attention mechanism to improve the classification performance. Liu and Zhang [16] extended the attention modeling by distinguishing the attention obtained from the left context and the right context of a given target or aspect, and further controlled the role of attention by adding multiple gates. Tang et al. [17] introduced an end-to-end memory network for aspect-level sentiment classification, which employs an attention mechanism with external memory to capture the importance of each context word with respect to a given target aspect. Ma et al. [18]

proposed an Interactive Attention Network (IAN) model based on LSTM and attention mechanism, which modeled the target word and context text separately and made them interact after the LSTM processing. Chen et al. [19] believed that sentiment analysis of complex sentences is difficult, and subtle semantic features cannot be easily captured, so they adopted multi-layer attention mechanism to capture the relationship between distant words in the sentence. With the successive proposals of contextual-dependent representation models like ELMo [20], BERT [21], XLNET [22] and RoBERTa [23], they have achieved better results in ALSC than traditional deep learning models.

Prompt tuning

As a new fine-tuning paradigm, prompt tuning was proposed to bridge the objective gap between pre-training and fine-tuning. With appropriate prompts and tuning goals, prompt tuning can manipulate model behavior to adapt to a variety of downstream tasks. By using specially constructed prompts, we can further inject and activate task-related knowledge into the PLMs (Pretrained Language Models), thereby improving the task-specific performance of the model. However, making suitable ALSC prompts by hand requires domain expertise, and automatically constructing a well-performing prompt usually requires additional computational costs to verify.

Schick et al. [24] believed that the effect of traditional supervised learning method would be unsatisfactory in a few-shot setting, because they are insufficient to finetune the model to fully understand the task. On the other hand, appropriate textual explanation can help the model understand what the task is. Moreover, the same approach can be applied to any language model, including GPT [25], BERT [21], Seq2Seq, etc. So, they introduced Pattern Exploiting Training (PET) method to convert input samples into cloze-style type text to help language model better understand tasks. They proposed an unsupervised PET method of iterative training called iPET. Gao et al. [26] believed that GPT-3 [27] achieved strong performance by fine tuning on small samples, but its large size limited application in more scenes. Inspired by GPT-3, they proposed a LM-BFF (Better Few-shot Fine-tuning of Language Model), which included: (1) a fine-tuning method based on prompt, and the method of automatically generating prompt template; (2) dynamic selection of samples. Zhang et al. [28] used certain parameters in the language model as templates and tags, and optimized them through back propagation without introducing other parameters outside the model. They proposed a DART fine-tuning method. Han et al. [29] suggested that prompt methods had shown significant improvements in a number of few-shot text understanding tasks (sentiment analysis/semantic reasoning). Manually building natural language prompts could be cumbersome and error-prone, and validation was time-consuming for automatically generated prompts. So, they used prompt tuning with rules (PTR) to perform tuning tasks for multiple categories. Li et al. [9] proposed SentiPrompt-tuning. Given known aspect and opinion, a continuous template could be constructed to predict the corresponding sentiment polarity categories. This was the earliest known work in which prompt was used for aspect-level sentiment analysis tasks. Seoh et al. [10] made an open aspect target emotion classification based on natural language prompts. In our model, we follow the idea of previous work and build templates to better fit the pre-training model. Unlike the common practice, we directly use masked hidden vectors for classification.

BERT-based methods

The Encoder block of the transformer model consists of a self-attention layer, a residual connection and normalization layer, and a feedforward neural network layer. As representative models in the Encoder-only Transformer family, BERT [21] and RoBERTa [23] show outstanding performance in natural language understanding tasks. They are different from traditional RNN and CNN models, reducing the semantic distance between any word pair through the attention mechanism. It effectively solves the intractable long-term dependency problem in natural language processing. The context can also be considered from both sides of the word. This bidirectionality helps the model better understand the context in which the word is used.

BERT consists of two pre-training tasks, one is masked language modeling and the other is next sentence prediction. The first task is converted into a cloze problem. It randomly masks 15% of the words in each sentence, then tries to determine what the masked word should have been with its context. GPT uses the traditional language modeling objective, which restricts the context to be unidirectional, but BERT can extract contextual information from both sides. The second task builds sentence pairs from an unlabeled collection, and the model predicts whether the pair are valid consecutive sentences in the original text. RoBERTa is an improved version of BERT with many design revisions, more training data, and the elimination of the next sentence prediction task.

Song et al. [30] converted aspect entities and contexts into sentence pair classification tasks and input them into the downstream modules of BERT to facilitate the judgment of the emotional polarity of aspect entities. Gao et al. [31] changed the default output location at CLS when using BERT results for classification. They used the positions for various aspects and show clear improvement over similar approaches. Li et al. [32] directly treated the task as sequence labeling. They used BERT as the context encoder, and then carried out experiments with a variety of downstream models on the BERT output, including Linear/GRU/Self-Attention/CRF. Sun et al. [33] used BERT to perform aspect-based sentiment analysis by constructing auxiliary sentences. Phan et al. [34] believed that the previous approaches to the ALSC task were unable to explain the grammatical correlation between aspect words and contextual words. They explored the grammatical aspect words in sentences and applied the self-attention mechanism to semantic learning.

Tian et al. [35] introduced a novel graph convolutional network that utilizes an attention mechanism to differentiate the semantic relationships between different words. This method combines the semantic information learned by different GCN layers through an attention layer and applies this approach to BERT. Liang et al. [36] proposed a novel framework, the Bi-Syntax aware Graph Attention Network (BiSyn-GAT+), which effectively leverages syntactic information by modeling both intra-context and inter-context information in the form of trees. Zhang et al. [37] introduced a novel Syntactic and Semantic Enhanced Graph Convolution Network (SSEGCN) model. Specifically, they proposed an Aspect-aware Attention Mechanism with self-attention to obtain an attention score matrix for sentences, and they applied this approach to BERT.

Yang et al. [38] referenced the model structure of ConvS2S [39], which utilizes convolution to extract local feature information from source code. Sentence S, with an attribute word A, is referred to as the global context when represented

as [CLS]+S+[SEP]+A+[SEP], and as the local context when represented as [CLS]+S+[SEP]. Zeng et al. [40] employs this method to extract both global and local semantic information. The difference in our approach is that we combine information extracted from pre-trained models BERT/RoBERTa with information extracted through convolutional methods. Currently, there is no aspect-level sentiment analysis approach that utilizes this methodology. Additionally, we utilize the masking mechanism of BERT/RoBERTa to create hard prompt templates for the dataset. Diverging from previous practices, we directly extract vectors from the merged feature vectors at the positions marked with [Mask] for sentiment classification.

Method

Problem definition and notations

ALSC is the problem of determining it's the sentiment polarity for each aspect in a sentence. Suppose we have input words $X = \{x_1, \cdots, x_T\}$, in which the aspect words are $A = \{a_1, \cdots, a_K\}$, and the aspect sentiment classification is a {positive, neutral, negative} label for each of the target in the sentence. For example, given the sentence "The hamburger is delicious, but the service is terrible.", it contains two aspect terms, hamburger and service. The goal of the task is to analyze the sentiment polarity corresponding to each of them. For the example above, input is the sentence and the aspect words, and the output is the sentiment polarity for the aspect words. Figure 1 shows the input and output of ALSC task with an example.

Model description

We propose an PConvBERT (Prompt-ConvBERT) and an PConvRoBERTa (Prompt-ConvRoBERTa) model. They have the same structure with BERT or RoBERTa as the Transformer backbone.

As shown in Fig. 2, the input data is divided into text content and sentiment polarity labels. Cloze-type prompts are composed with aspect terms appended to input text. Downstream structure includes a word embedding layer with adversarial training FGM (Fast Gradient Method) [41], Transformer Encoder, LSFE (Local Semantic Feature Extractor) and Focal Loss. Specially, we use the method FGM to perturb the word embedding layer of the BERT/RoBERTa to improve the robustness of the model. The whole pipeline is trained in an end-to-end manner to obtain an ALSC model. In the

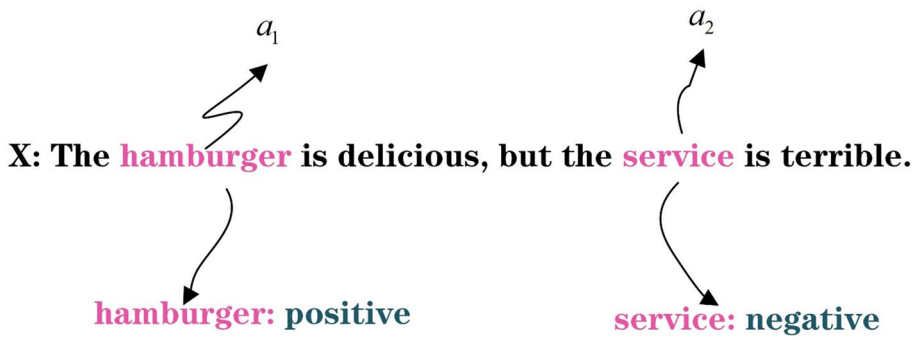

Fig. 1 Example of a customer review. It contains two aspect words, hamburger and service, while the sentiment polarity towards *hamburger* is positive and that of *service* is negative

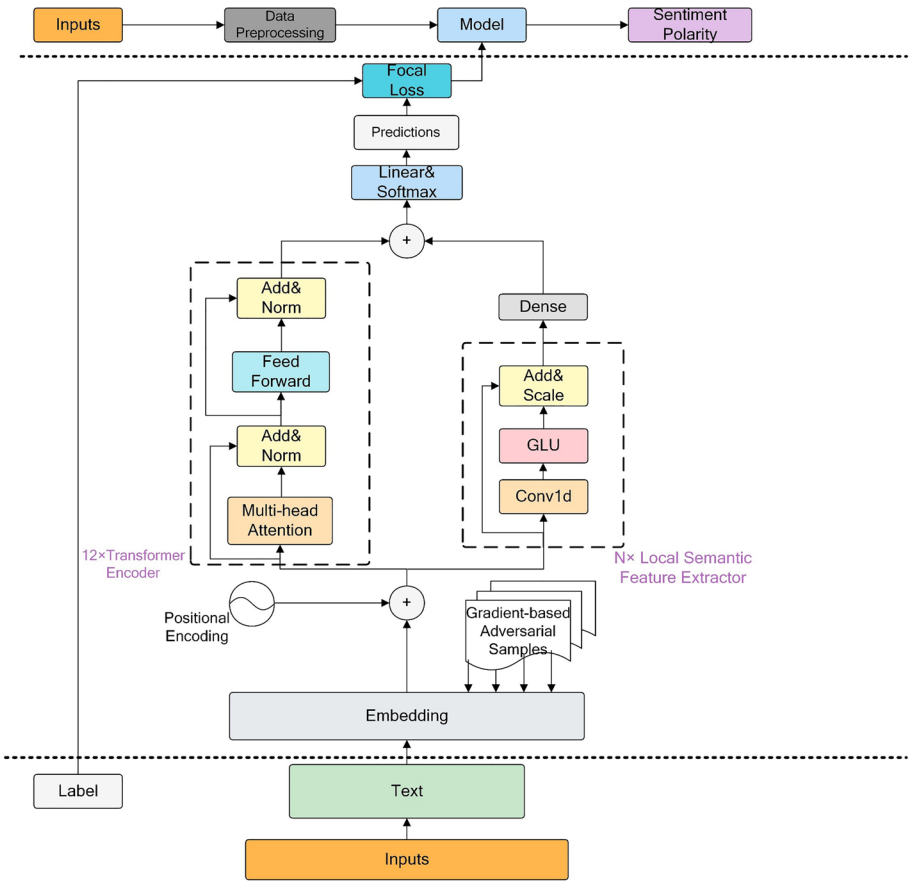

Fig. 2 The overall architecture of PConvBERT (RoBERTa)

data preprocessing stage, the first step is to extract essential information from the raw data. Then, sentences are constructed using templates. Subsequently, tokenization is performed, with BERT utilizing WordPiece as its tokenizer, while RoBERTa uses BBPE (Byte-level Byte-Pair Encoding) as its tokenizer. Finally, each subword is mapped to its corresponding index sequence in the vocabulary, which serves as the input to the model. The input text is fed into the model after data preprocessing. Our model is trained on the processed data, and after merging the feature information extracted from BERT/RoBERTa and LSFE, the feature vector at the [Mask] position is taken and fed into linear and softmax layers for classification.

Prompt

To better utilize the pretrained language model, two different types of Prompts are designed, hard prompt and soft prompt type. Both of these methods transform the input form of downstream tasks to fit the pre-trained model.

The hard prompt type methods refer to manually creating templates for input data. If the original sentence is "god of war psp is very satisfying.", it is likely converted into "The sentence is 'god of war psp is very satisfying', where psp means [Mask].". Such templates can better explore the potential of pre-trained models.

To better adapt the downstream tasks to the pre-trained model, we come up with the following set of cloze-type hard-prompts that are aspect dependent and apply them to input text.

The sentence is '{sentence}', where {aspect} means [Mask].
The sentence is '{sentence}', where I felt the {aspect} was [Mask].
The sentence is '{sentence}', where the {aspect} made me feel [Mask].
The sentence is '{sentence}', where the {aspect} is [Mask].

In the templates above, {sentence} is the placeholder for the whole original input sentence, {aspect} is the placeholder for the querying aspect term, and [Mask] represents the masked word for BERT/RoBERTa, with slightly different specifications. We extract feature vector from the [Mask] position for sentiment classification. To measure if the design of prompts helps improve performance, we compare the same model with or without templates in the ablation experiment.

LSFE

Recently, research in aspect-level sentiment classification places more attention on the aspect-related semantic information in a sentence. How to retain complete semantic information to judge the sentiment polarity of aspect words has become a concern of researchers (Liang et al. [36]). For ALSC, the semantic features of sentences are very important, but the self-attention mechanism in BERT and RoBERTa, focuses more on the global semantic features. Therefore, we add a Local Semantic Feature Extractor (LSFE) to place more attention on the local semantic information, for which a limited visible window is more desirable. So, we insert a CNN component that complements BERT or RoBERTa in ALSC.

As shown in Fig. 2, LSFE consists of a one-dimensional convolution, Gated Linear Unit (GLU), residual connection and scaling. One-dimensional convolution mainly extracts the features of the sentence. After convolution and GLU activation, the output is added to the input with residual connection and then scaled to the designed size.

The input vector X (i.e., the vector X obtained through the word embedding layer and positional encoding) is fed to both the multi-layer bidirectional encoder (BERT/RoBERTa) and multi-layer LSFE to extract both the global and local semantic information. S_c denotes the scaled value in LSFE layer output. After completing the local semantic feature extraction in LSFE, a dense layer is used to reshape the feature vector in order to obtain the feature vector H, so that it fits the dimension of context vector F from the BERT/RoBERTa. The two features are merged through matrix addition, and the context representation vector Z is obtained. Subsequently, feature vectors $Z_{[Mask]}(Z_{[Mask]} \in R^{batch*hidden\_size})$ are extracted from the [Mask] positions in vector Z. Finally, the extracted feature vectors $Z_{[Mask]}$ are fed into a linear layer and the softmax function for predicting sentiment labels.

$$H = Dense((X + GLU(Conv(X))) * S_c), \tag{1}$$

$$Z = F + H, \tag{2}$$

$$P(t) = softmax\left(W_p Z_{[Mask]} + b_p\right). \tag{3}$$

FGM and loss

Many aspect-level sentiment classification models suffer from stability issues, so we also add adversarial training FGM [41] to improve the robustness of the model.

Choice of perturbation:

$$r_{adv} = -\epsilon g/||g||_2 where\, g = \nabla_x \log p\left(y|x;\widehat{\theta}\right). \tag{4}$$

And adversarial examples include the perturbation factor for better model stability:

$$x_{adv} = x + r_{adv}. \tag{5}$$

The loss function is minimized with the additional noise, which improves the robustness of the model. Here g is the gradient of the loss function, so we take the $-g$ as the update direction. ϵ can be viewed as a value that regulates the size of the perturbation.

Assuming a binary classification scenario, a standard cross entropy can be written as:

$$p_t = \begin{cases} p & if\ y = 1 \\ 1 - p & otherwise \end{cases}. \tag{6}$$

$$CE\left(p, y\right) = CE(p_t) = -\log(p_t). \tag{7}$$

To solve the imbalance of positive and negative samples, a common practice is to add weighting factors to form balanced cross entropy. We add α to class 1 and 1-α to class -1 under the premise that $\alpha \in [0, 1]$. We adopt α_t uniformly.

$$CE(p_t) = -\alpha_t \log(p_t). \tag{8}$$

In order to distinguish difficult samples and easy samples, Lin et al. [42] added the modulating factor $(1 - p_t)^\gamma$ with tunable parameter $\gamma \geq 0$. They defined the focal loss as

$$FL(p_t) = -(1 - p_t)^\gamma \log(p_t). \tag{9}$$

To balance the problem of positive and negative samples at the same time, they also combined it with the weighted cross-entropy loss, so the final equation of Focal Loss is:

$$FL(p_t) = -\alpha_t (1 - p_t)^\gamma \log(p_t). \tag{10}$$

The focal loss is used to calculate the disagreement between the predicted label and the true label. We set γ=2 in our experiments.

Experiments

Datasets

We carry out the experiment on the widely used SemEval-2014 task4 [43], Twitter [44] and MAMS [45]. Sentiment labels include positive, neutral and negative for each aspect. Table 1 shows the details of the dataset.

Table 1 Data sample distribution

Datasets	Pos	Neu	Neg
Laptop			
Train	987	460	866
Test	341	169	128
Restaurant			
Train	2164	633	805
Test	728	196	196
Twitter			
Train	1561	3127	1560
Test	173	346	173
MAMS			
Train	3380	5042	2764
Dev	403	604	325
Test	400	607	329

Table 2 Hyper-parameter setting

Parameter	Value
Dropout rate	0.1
Batch size	32
Learning rate	2e−5
Max epoch	35
Early stopping	6

Experiment settings

In our experiments, we use PyTorch to implement all the models and fine-tune $BERT_{base}$ and $RoBERTa_{base}$ on a single NVIDIA RTX 3090 GPU. The number of transformer layers is fixed at 12, size of the hidden layer is 768, and the number of self-attention heads is 12. We use the AdamW optimizer for model tuning, and other hyper-parameters in the experiment are shown in Table 2.

Baseline methods

We use classification accuracy and Macro-F1 to compare the performance of our model to previous methods. Baseline methods used in the experiment are shown below:

TD-LSTM [13]: The sentence is divided into two parts, context preceding the aspect term and context following the aspect. Each of the two parts are processed by a unidirectional LSTM and concatenated before the softmax classifier.

IAN [18]: Two LSTMs are used to model the aspect entity and the context respectively, and then the respective feature representations interact through the attention mechanism.

MemNet [17]: The context information is used to construct a memory network, and attention is used to capture the related information for different aspects.

RAM [19]: After the sentence is input into bidirectional LSTM, a multi-layer attention mechanism is used to synthesize the important features in the sentence.

BERT-SPC [30]: The aspect entities and contexts are transformed into a sentence pair classification task.

BERT-PT [46]: BERT is finetuned for a review reading comprehension task, and aspect-level sentiment analysis is solved with that finetuned task.

BAT [47]: A generic BERT model is finetuned on domain-specific data with adversarial training.

T-GCN [35]: It utilizes dependency types to distinguish different relations in the graph and uses attentive layer ensemble to learn the contextual information from different GCN layers.

dotGCN [8]: A model for automatically inducing discrete opinion trees for each aspect.

BERT [21]: This is the base BERT model without any task-specific revision.

RoBERTa [23]: An updated version of BERT, with revisions in model design, training data and pretraining procedure.

Experimental and analysis

Main results

The experimental results are shown in Table 3, in which our proposed model outperforms all baseline methods on Restaurant, Laptop and MAMS. As can be seen from Table 3, Transformer models have clear advantage over word embedding models. This indicates that the pre-trained language model has greater ability of language representation and feature extraction with its context-dependent representation. Compared to dotGCN + BERT, PConvBERT exhibited an average improvement of 0.71% in accuracy and 0.30% in Macro-F1 on the Restaurant and Laptop datasets. However, on the Twitter dataset, PConvBERT showed a decrease of 1.38% in accuracy and 1.18% in Macro-F1 compared to dotGCN + BERT. The possible reason for this occurrence is that there

Table 3 Comparison of classification accuracy and Macro-F1 for ALSC

Methods	Restaurant		Laptop		Twitter		MAMS	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
TD-LSTM	75.63	–	68.13	–	70.80	69.00	–	–
IAN	78.60	–	72.10	–	–	–	–	–
MemNet	78.16	65.83	70.33	64.09	–	–	–	–
RAM	80.23	70.80	74.49	71.35	69.36	67.30	–	–
BERT	82.86	74.87	77.12	72.55	74.42	72.67	81.96	81.28
BERT-SPC	84.46	76.98	78.99	75.03	73.55	72.14	82.82	81.90
BERT-PT	84.95	76.96	78.07	75.08	–	–	–	–
BAT	86.03	79.24	79.35	76.50	–	–	–	–
T-GCN + BERT	86.16	79.95	80.88	77.03	76.45	75.25	83.38	82.77
dotGCN + BERT	86.16	80.49	81.03	78.10	**78.11**	**77.00**	–	–
PConvBERT (ours)	**86.96**	**80.87**	**81.66**	**78.33**	76.73	75.82	**84.36**	**83.95**
RoBERTa	87.23	80.20	81.19	77.69	74.58	72.75	84.06	83.45
PConvRoBERTa (ours)	**89.29**	**84.27**	**83.54**	**80.89**	**78.47**	**77.53**	**85.55**	**85.05**

All results of baseline models, except BERT and RoBERTa, are retrieved from original publication. "–" means not reported

are a lot of colloquial and informal sentences in the comments on Twitter dataset, which diverge from standard written language. The model in this study may have limited proficiency in processing this type of sentence. It is worth noting that PConvBERT outperformed T-GCN + BERT, the third-best BERT-based model on average, with an improvement of 0.63% in accuracy and 0.88% in Macro-F1 on the Twitter and MAMS datasets. The best performance in each column is bold-typed.

To verify if a better baseline would offset the improvement, we also consider RoBERTa, which is known for its performance boost over BERT. Our PConvRoBERTa model also achieves competitive performance on both datasets. PConvRoBERTa has an average accuracy increase of 2.45% compared to RoBERTa, while the average Macro-F1 has increased by 3.41%. In all the compared models, PConvRoBERTa achieves the best results on Restaurant, Laptop, Twitter and MAMS.

Ablation study

To verify the value of our design choices, a controlled experiment is carried out to tune the PConvBERT and PConvRoBERTa model with certain components removed, where w/o means that the related component is not applied in the corresponding version. As shown in Table 4, We analyzed the performance of PConvBERT and PConvRoBERTa with each of the four components removed, LSFE, Focal Loss, FGM, and Prompt.

It can be seen from the experimental results that PConvBERT and PConvRoBERTa have a large performance gap over BERT and RoBERTa. PConvBERT has average improvement of 3.34% accuracy and 4.4% Macro-F1 compared with BERT on 4 datasets. And PConvRoBERTa obtains an average of + 2.45% accuracy and + 3.41% Macro-F1 over RoBERTa on 4 datasets. However, the performance of PConvRoBERTa is decreased without any of the four components including LSFE (− 1.68% accuracy and − 2.3%

Table 4 Results of ablation study

Methods	Restaurant		Laptop		Twitter		MAMS	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
PConvBERT	**86.96**	**80.87**	**81.66**	**78.33**	**76.73**	**75.82**	**84.36**	**83.95**
PConvBERT w/o LSFE	85.36	78.70	79.15	74.87	74.86	74.08	83.08	82.52
PConvBERT w/o Focal Loss	85.36	78.15	79.31	75.34	74.46	72.83	83.76	83.26
PConvBERT w/o FGM	86.34	79.35	80.56	77.83	75.00	73.92	82.93	82.48
PConvBERT w/o Prompt	85.36	79.13	76.96	70.74	75.43	74.24	83.08	82.53
P ConvRoBERTa	**89.29**	**84.27**	**83.54**	**80.89**	**78.47**	**77.53**	**85.55**	**85.05**
PConvRoBERTa w/o LSFE	87.86	80.65	82.29	79.31	76.01	75.00	83.98	83.57
P ConvRoBERTa w/o Focal Loss	85.62	77.97	82.45	79.72	76.30	75.28	83.23	82.62
PConvRoBERTa w/o FGM	87.77	81.56	82.92	80.11	75.87	75.04	83.01	82.44
PConvRoBERTa w/o Prompt	86.43	78.70	83.07	80.13	73.84	72.84	84.88	84.35

The best performance in each column is bold-typed

Macro-F1), Focal Loss (− 2.31% accuracy and − 3.04% Macro-F1), FGM (− 1.82% accuracy and − 2.15% Macro-F1) or Prompt (− 2.16% accuracy and − 2.93% Macro-F1). Like PConvRoBERTa, the PConvBERT model also shows performance drop without one of the components.

Ablation experiments above show that the each of the design components in PConvBERT and PConvRoBERTa are necessary, as the missing of any component clearly hurts performance.

Prompt analysis

For pretrained language model, the right prompt can accurately activate the task-specific knowledge, resulting in the best performance. For comparison, we have tried four templates for the input text, and the accuracy of each template on 4 datasets are shown in Table 5.

Choice of hyper-parameters

The choice of hyper-parameters also has a large impact on model performance. Figures 3 and 4 shows the number layers of LSFE (n_layers) and kernel size, where number of layers are experimented in the 1–12 range, and kernel size takes [3, 5, 7]. Since the training process with FGM is relatively unstable, we choose to run the experiment without the FGM component. As can be seen from Fig. 3, PConvRoBERTa w/o FGM can achieve the highest accuracy of 88.04% on restaurant, and the approximate accuracy range is between 87 and 88%. From Fig. 4, and the approximate accuracy range is between 81% and 82.5% for Laptop, but both figures show a wide range of performance without a clear pattern.

As shown in Tables 6 and 7, when the best choice of LSFE layers and kernel size is take from the above experiment, perturbation rate ε in FGM is another hyper-parameter that affects the model performance. On the Restaurant dataset, the convolutional kernel size is 3, and the layers of LSFE are set to 5. Meanwhile, on the Laptop dataset, the convolutional kernel size is 3, and the layers of LSFE are set to 1. Referring to the method of adversarial training for BERT proposed by Karimi et al. [47], the disturbance rate ε value

Table 5 A comparison of four templates on four datasets

Template	Models	Restaurant Accuracy	Laptop Accuracy	Twitter Accuracy	MAMS Accuracy
The sentence is '{sentence}', where {aspect} means [Mask]	PConvBERT	**86.96**	**81.66**	**76.73**	**84.36**
	PConvRoBERTa	**89.29**	**83.54**	**78.47**	84.96
The sentence is '{sentence}', where I felt the {aspect} was [Mask]	PConvBERT	86.43	78.21	75.00	83.08
	PConvRoBERTa	87.05	81.66	76.45	85.10
The sentence is '{sentence}', where the {aspect} made me feel [Mask]	PConvBERT	85.80	80.88	73.99	83.08
	PConvRoBERTa	86.52	82.13	75.29	84.21
The sentence is '{sentence}', where the {aspect} is [Mask]	PConvBERT	85.62	79.15	75.14	83.98
	PConvRoBERTa	88.39	82.45	74.86	**85.55**
–	PConvBERT	85.36	76.96	75.43	83.08
	PConvRoBERTa	86.43	83.07	73.84	84.88

The best performance in each column is bold-typed

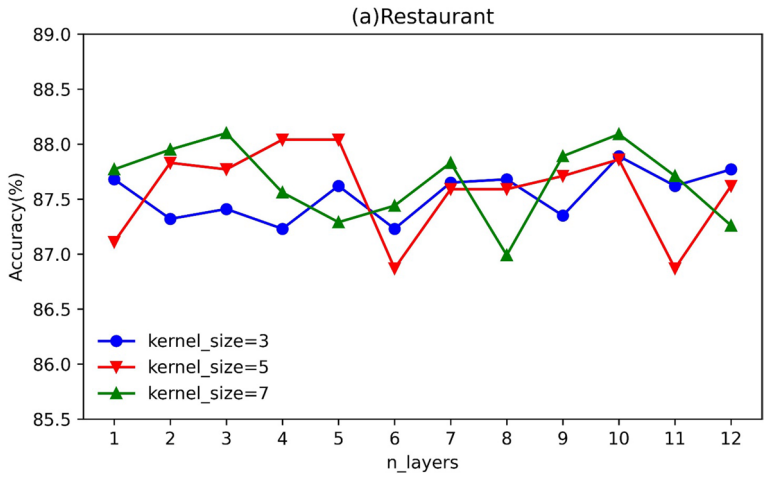

Fig. 3 Average accuracy for PConvRoBERTa on restaurant with different of n_layers and kernel size value

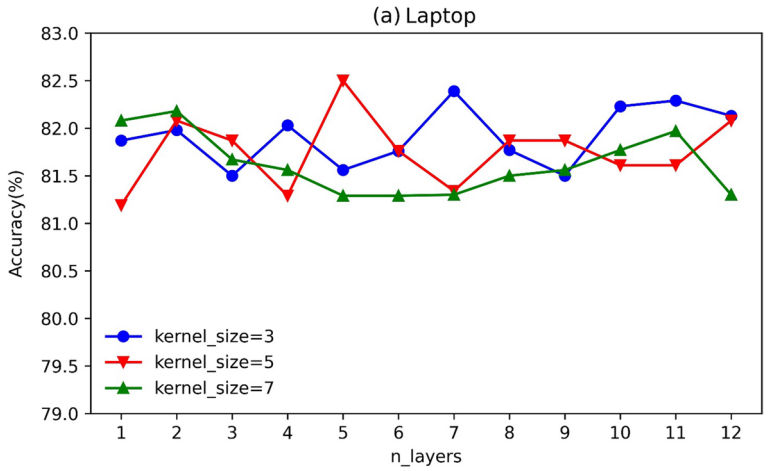

Fig. 4 Average accuracy for PConvRoBERTa on laptop with different of n_layers and kernel size value

Table 6 Performance of PConvRoBERTa on restaurant with different perturbation rate ε

Models	Restaurant	
	Accuracy	**Macro-F1**
PConvRoBERTa (ε = 0.01)	87.77	82.15
PConvRoBERTa (ε = 0.1)	87.86	81.07
PConvRoBERTa (ε = 1)	87.77	81.90
PConvRoBERTa (ε = 2)	87.86	82.51
PConvRoBERTa (ε = 5)	**89.29**	**84.27**

The best performance in each column is bold-typed

is set to {0.01, 0.1, 1, 2, 5}. It can be seen from the 2 tables that PConvRoBERTa performs better when the perturbation rate is relatively large, but the optimal value is different for the two datasets.

Table 7 Performance of PConvRoBERTa on Laptop with different perturbation rate ε

Models	Laptop	
	Accuracy	**Macro-F1**
PConvRoBERTa (ε = 0.01)	82.13	79.16
PConvRoBERTa (ε = 0.1)	81.35	78.21
PConvRoBERTa (ε = 1)	82.13	78.95
PConvRoBERTa (ε = 2)	**83.54**	**80.89**
PConvRoBERTa (ε = 5)	82.13	78.75

The best performance in each column is bold-typed

Conclusion and future work

Both BERT and RoBERTa adopt self-attention mechanism to captures global semantic information from the context. As its supplement, we also include an LSFE module to capture local semantic information. Moreover, the Aspect-Level Sentiment Classification task suffers from stability issues, so we introduce adversarial training to improve the robustness of our model, which effectively improves the classification performance. And to deal with the problem of unbalanced sample distribution, Focal Loss is applied. We use natural language prompts to maximize the ability of the pre-trained models. Specially, we utilize [Mask] vector outputs of templates for sentiment classification. The above methods constitute PConvBERT and PConvRoBERTa. Experiments show that our method achieves significant improvements over other methods on multiple datasets. Moreover, ablation study displays the necessity for all design choices, where the removal of any component degrades performance. In the future, more complex attention mechanism with external knowledge [48] and task-specific tuning on pre-trained models can be considered to improve the effectiveness of our proposed model. The attention mechanism may also be applied as a filter to first identify the important parts of the sentence or external memory to improve the accuracy. Compression methods such as knowledge distillation can be investigated to make the model more compact and efficient. Such a model may even work on less computation-intensive devices. All these are worth further research to improve the performance or availability of the model.

Acknowledgements
Not applicable.

Author contributions
AF provided the research idea and resources for this work, and also revised the manuscript for multiple times. JC conducted the experiments and wrote the first draft. ZG and XL provided comments and guided experiments for the revised version.

Funding
The work was supported by Sichuan Science and Technology program with award number 2023YFS0453.

Availability of data and materials
All data supporting this systematic review are from previously reported studies and datasets, which have been cited within the article. The processed data are available from the corresponding author upon request.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 10 April 2023 Accepted: 4 December 2023
Published online: 19 December 2023

References

1. Bakshi RK, Kaur N, Kaur R, Kaur G. Opinion mining and sentiment analysis. In: 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom). 2016; p. 452–55.
2. Zhang D, Zhu Z, Kang S, Zhang G, Liu P. Syntactic and semantic analysis network for aspect-level sentiment classification. Appl Intell. 2021;51:6136–47.
3. Zhang W, Li X, Deng Y, Bing L, Lam W. A survey on aspect-based sentiment analysis: tasks, methods, and challenges. 2022.
4. Nazir A, Rao Y, Wu L, Sun L. Issues and challenges of aspect-based sentiment analysis: a comprehensive survey. IEEE Trans Affect Comput. 2020;13:845.
5. Luo H, Ji L, Li T, Jiang D, Duan N. Grace: gradient harmonized and cascaded labeling for aspect-based sentiment analysis. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings. 2020; p. 54–64.
6. Ma F, Zhang C, Song D. Exploiting position bias for robust aspect sentiment classification. In: ACL. 2021; p. 1352–58.
7. Oh S, Lee D, Whang T, Park I, Seo G, Kim E, Kim H. Deep context- and relation-aware learning for aspect-based sentiment analysis. In: Annual Meeting of the Association for Computational Linguistics. 2021.
8. Chen C, Teng Z, Wang Z, Zhang Y. Discrete opinion tree induction for aspect-based sentiment analysis. In: Annual Meeting of the Association for Computational Linguistics. 2022.
9. Li C, Gao F, Bu J, Xu L, Chen X, Gu Y, Shao Z, Zheng Q, Zhang N, Wang Y, Yu Z. SentiPrompt: sentiment knowledge enhanced prompt-tuning for aspect-based sentiment analysis. 2021; arXiv, abs/2109.08306.
10. Seoh R, Birle I, Tak M, Chang H, Pinette B, Hough A. Open aspect target sentiment classification with natural language prompts. In: Conference on Empirical Methods in Natural Language Processing. 2021.
11. Dietterich TG. Machine learning. ACM Comput Surv. 1996;28:3.
12. Huang B, Carley KM. Parameterized convolutional neural networks for aspect level sentiment classification. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, October-November 2018, Brussels. 2019; 1091–1096. https://doi.org/10.18653/v1/D18-1136.
13. Tang D, Qin B, Feng X, et al. Effective LSTMs for target-dependent sentiment classification. 2015.
14. Wang Y, Huang M, Zhao L. Attention-based LSTM for aspect-level sentiment classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, November 2016; p. 606–15. https://doi.org/10.18653/v1/D16-1058.
15. Yang M, Tu W, Wang J, et al. Attention based LSTM for target dependent sentiment classification. In: Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, California, 4–9 February 2017; p. 5013–14.
16. Liu J, Zhang Y. Attention modeling for targeted sentiment. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Vol. 2, Short Papers, 2017; p. 572–77. https://doi.org/10.18653/v1/E17-2091.
17. Tang D, Qin B, Liu T. Aspect level sentiment classification with deep memory network. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, November 2016, p. 214–24. https://doi.org/10.18653/v1/D16-1021.
18. Ma D, Li S, Zhang X, et al. Interactive attention networks for aspect-level sentiment classification. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence Main Track, Melbourne, 19–25 August 2017; p. 4068–74. https://doi.org/10.24963/ijcai.2017/568.
19. Chen P, Sun Z, Bing L, Yang W. Recurrent Attention Network on Memory for Aspect Sentiment Analysis. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark. Association for Computational Linguistics. 2017; p. 452–61.
20. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L. 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), New Orleans, Louisiana. Association for Computational Linguistics. 2018; p. 2227–37.
21. Devlin J, Chang M, Lee K, Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. ArXiv, abs/1810.04805. 2019.
22. Yang Z, Dai Z, Yang Y, Carbonell J, Salakhutdinov RR, Le QV. XLNet: Generalized autoregressive pretraining for language understanding. In: Proc. Adv. Neural Inf. Process. Syst., vol. 32, 2019, p. 5753–63.
23. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V. RoBERTa: A robustly optimized BERT pretraining approach. 2019, arXiv:1907.11692.
24. Schick T, Schütze H. Exploiting cloze-questions for few-shot text classification and natural language inference. In: Conference of the European Chapter of the Association for Computational Linguistics. 2020.
25. Radford A, Narasimhan K. Improving language understanding by generative pre-training. 2018.
26. Gao T, Fisch A, Chen D. Making pre-trained language models better few-shot learners. ArXiv, abs/2012.15723. 2021.
27. Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, Agarwal S, Herbert-Voss A, Krueger G, Henighan TJ, Child R, Ramesh A, Ziegler DM, Wu J, Winter C, Hesse C, Chen M, Sigler E, Litwin

M, Gray S, Chess B, Clark J, Berner C, McCandlish S, Radford A, Sutskever I, Amodei D. Language models are few-shot learners. ArXiv, abs/2005.14165. 2020.
28. Zhang N, Li L, Chen X, Deng S, Bi Z, Tan C, Huang F, Chen H. Differentiable prompt makes pre-trained language models better few-shot learners. ArXiv, abs/2108.13161. 2021.
29. Han X, Zhao W, Ding N, Liu Z, Sun M. PTR: prompt tuning with rules for text classification. AI Open. 2021;3:182–92.
30. Song Y, Wang J, Jiang T, et al. Attentional encoder net-work for targeted sentiment classification. 2019.
31. Gao Z, Feng A, Song X, et al. Target-dependent sentiment classification with BERT. IEEE Access. 2019;7:154290–9. https://doi.org/10.1109/ACCESS.2019.2946594.
32. Li X, Bing L, Zhang W, Lam W. Exploiting BERT for end-to-end aspect-based sentiment analysis. association for computational linguistics. In: Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019), 2019; p. 34–41. https://doi.org/10.18653/v1/D19-5505.
33. Sun C, Huang L, Qiu X. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. 2019
34. Phan MH, Ogunbona P. Modelling context and syntactical features for aspect-based sentiment analysis. In: Annual Meeting of the Association for Computational Linguistics. 2020.
35. Tian Y, Chen G, Song Y. Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble. In: North American Chapter of the Association for Computational Linguistics. 2021.
36. Liang S, Wei W, Mao X, Wang F, He Z. BiSyn-GAT+: bi-syntax aware graph attention network for aspect-based sentiment analysis. Findings. 2022
37. Zhang Z, Zhou Z, Wang Y. SSEGCN: syntactic and semantic enhanced graph convolutional network for aspect-based sentiment analysis. In: North American Chapter of the Association for Computational Linguistics. 2022.
38. Yang G, Zhou Y, Chen X, Yu C. Fine-grained Pseudo-code Generation Method via Code Feature Extraction and Transformer. In: 2021 28th Asia-Pacific Software Engineering Conference (APSEC), 2021, p. 213–22. https://doi.org/10.1109/APSEC53868.2021.00029.
39. Gehring J, Auli M, Grangier D, Yarats D, Dauphin YN. Convolutional sequence to sequence learning. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70 (ICML'17). JMLR.org. 2017; p. 1243–52.
40. Zeng B, Yang H, Xu R, Zhou W, Han X. LCF: a local context focus mechanism for aspect-based sentiment classification. Appl Sci. 2019.
41. Miyato T, Dai AM, Goodfellow IJ. Adversarial training methods for semi-supervised text classification. arXiv: Machine Learning. 2016.
42. Lin T-Y, Goyal P, Girshick R, He K, Dollár P. Focal Loss for Dense Object Detection. IEEE Int Conf Comput Vision (ICCV). 2017;2017:2999–3007. https://doi.org/10.1109/ICCV.2017.324.
43. Pontiki M, Galanis D, Pavlopoulos J, Papageorgiou H, Androutsopoulos I, Manandhar S. Semeval-2014 task 4: aspect based sentiment analysis. In: Proceedings of the 8thInternational Workshop on Semantic Evaluation (SemEval2014), Association for Computational Linguistics, Dublin, Ireland, August 2014. p. 27–35.
44. Dong L, Wei F, Tan C, Tang D, Zhou M, Xu K. Adaptive recursive neural network for target-dependent twitter sentiment classification. In: Annual Meeting of the Association for Computational Linguistics. 2014.
45. Jiang Q, Chen L, Xu R, Ao X, Yang M. A challenge dataset and effective models for aspect-based sentiment analysis. In: Conference on Empirical Methods in Natural Language Processing. 2019.
46. Xu H, Liu B, Shu L, et al. BERT post-training for review reading comprehension and aspect-based sentiment analysis. In: Proceedings of NAACL-HLT 2019, Minneapolis, 2–7 June 2019, p. 2324–35.
47. Karimi A, Rossi L, Prati A. Adversarial training for aspect-based sentiment analysis with BERT. In: 2020 25th International Conference on Pattern Recognition (ICPR), 2021, p. 8797–03. https://doi.org/10.1109/ICPR48806.2021.9412167.
48. Chen Y, Zhuang T, Guo K. Memory network with hierarchical multi-head attention for aspect-based sentiment analysis. Appl Intell. 2021;51:4287–304.

Publisher's Note

Ao Feng received the B.E. and M.E. degrees in automation from Tsinghua University, China, in 1999 and 2001, respectively, and the M.S. and Ph.D. degrees in computer science from the University of Massachusetts Amherst, USA, in 2008. He has worked at Amazon.com and Lenovo research. He is currently an Associate Professor with the Department of Computer Science, Chengdu University of Information Technology. His research interests include information retrieval, data mining, natural language processing, and machine learning.

Jiazhi Cai received the B.E. degree from Chengdu College of University of Electronic Science and Technology of China in 2020. He is currently pursuing the M.E. degree in computer technology with the Chengdu University of Information Technology. His research interests include sentiment analysis, information extraction, and deep learning.

Zhengjie Gao received the B.E. and M.E. degree in computer science and technology from Chengdu University of Information Technology, China, in 2017 and 2020, respectively. His research interests include aspect-based sentiment classification and large language model.

Xiaojie Li received the Ph.D. degree in computer science and engineering from the School of Computer Science, Sichuan University, Chengdu, China, in 2015. She is a Professor with the Chengdu University of Information Technology, Chengdu. Her current research interests include machine learning, medical image segmentation, and deep learning.