Journal of Intelligent Information Systems (2023) 61:395–419
https://doi.org/10.1007/s10844-022-00760-2

Aspect-location attention networks for aspect-category sentiment analysis in social media

Pengfei Yu[1] · Wenan Tan[1,2] · Weinan Niu[1] · Bing Shi[1]

Received: 8 June 2022 / Revised: 20 October 2022 / Accepted: 20 October 2022 /
Published online: 14 December 2022

Abstract

As a fine-grained sentiment analysis, aspect-category sentiment classification aims to explore the implicit aspect information in text and analyze its sentiment polarity. When researching review data in social media, this task can often gain insight into the specific needs of users for a certain aspect of products, which is of great significance for commercial companies to improve their products. However, most aspect-level sentiment analysis targets aspect objects that appear directly in the text, which is limited in many scenarios. Furthermore, existing methods for aspect-category sentiment analysis rarely focus on the implicit location of aspect-category information in the context. To this end, the concept of Aspect-Location Attention Networks (ALAN) is proposed to integrate aspect-specific sentiment features for sentiment classification. In ALAN, a novel module is designed to differentially integrate aspect-category information into various locations of the context. The proposed models and their ablation models have been evaluated on three publicly available social review datasets, including two in English and one in Chinese. The experimental results show that ALAN and its variants outperform compared baseline models in terms of accuracy and macro F1-score.

Keywords Social media sentiment analysis · Aspect-category sentiment classification · Aspect-location embedding · Attention mechanism · BERT

✉ Wenan Tan
watan@sspu.edu.cn

Pengfei Yu
nuaaypf@nuaa.edu.cn

Weinan Niu
nuaanwn@nuaa.edu.cn

Bing Shi
bingshi@nuaa.edu.cn

[1] School of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, 211106, China

[2] School of Computer and Information Engineering, Shanghai Polytechnic University, Shanghai, 201209, China

1 Introduction

With the explosion of information in the era of big data, various social networks and short video platforms have generated enormous comment data. Users can give the most authentic and valuable reviews for specific products or services. The reviews effectively reflect the dynamic changes in user needs. Therefore, commercial companies need to identify the potential value of user reviews in a timely manner and improve their products accordingly to respond to market changes. Sentiment analysis on comment data is a good solution, but there is often more than one subject in a comment, which is not considered by traditional sentiment classification. The quality of a product can be judged by analyzing the sentiment polarity of a specific aspect in the review data. Hence, fine-grained sentiment analysis, specifically Aspect Category Sentiment Analysis (ACSA), has attracted widespread attention from researchers and businesses (Singh & Singh, 2021; Ozyurt & Akcayol, 2021), and is a hot topic in academic research.

The main research of sentiment analysis is to explore the sentiment polarity of texts with emotional overtones. Common sentiment polarity includes "positive", "negative", "neutral" and so on. When there are more than two sentiment polarities, it becomes a multi-classification problem and can be solved as a regression problem (Berka, 2020). According to the granularity of classification, sentiment classification can be divided into document-level, sentence-level, and aspect-level. While the first two emphasize the macro level, aspect-level sentiment analysis is more detailed and more demanding, requiring a correct judgment of the sentiment polarity of a specific aspect object in a text (Cambria, 2016). Moreover, ACSA is a major research direction in aspect-level sentiment classification tasks. For example, in the sentence "The pizza is very good and huge", the word "pizza" can be assigned to the aspect category "food". It can be judged as positive according to the semantics of the context. The aspect-category words may or may not appear directly in the text, or there may be more than one aspect-category. When inferring the sentiment of a given aspect, it is important to correctly analyze the contextual semantic features of the given aspect.

In recent years, methods based on deep neural networks for extracting text semantics have emerged and achieved good results for the ACSA tasks. To mine text features from the temporal information of text, schemes based on the long short-term memory model (LSTM Hochreiter et al. 1997) have also been continuously adopted. In addition, there are related studies that have noted many other types of rich structures between words and constructed graph networks for aspect-level sentiment analysis (Zhu et al., 2022). Although the results of these studies are effective, the question of how to add the feature information contained in aspect words to the context remains a worthwhile research problem. Most of the studies transform aspect words into a vector embedded in each word of the text. Although this method can retain the complete features of aspect words, it is not sufficiently targeted and may weaken the extraction effect of some key semantic features. In addition, since Bahdanau et al. (2015) proposed to apply attention mechanisms to the field of natural language processing, many researchers have also started to use attention mechanisms to capture important features of aspect words. Researchers have also devised many methods to compute the attention score, such as Self-Attention (Xiao et al., 2020), Hierarchical Attention (Geed et al., 2022), etc. Although most of the attention mechanisms are able to find sentiment words easily, they are more likely to focus on sentiment features that are not relevant to the aspect-category due to their inability to locate the given aspect-category. With this motivation, we pay more attention to the importance of aspect-category information at different locations in the context and propose "ALAN", an Aspect-Location

Attention Networks for ACSA. Our ALAN mainly consists of four modules. The first is the semantic representation module, which produces vector representations of words in sentences and aspect category words. The second is the proposed novel aspect-location embedding module. It differs from some approaches that combine aspect-category information indiscriminately because this module dynamically incorporates aspect-category information into the context, which enables the features that represent the aspect-category in the context to be more prominent. An improved attention mechanism is designed in the third module to focus on aspect-specific sentiment features. The attention network is different from traditional attention methods. In traditional methods, aspect category embeddings are often directly used as the benchmark for measuring attention scores, while the proposed method utilizes the results of the whole sentence combined with aspect-location embedding as the metric of attention. In this way, the attention weight can be calculated in a more targeted manner. The last module is the classifier, which is used to output features and perform sentiment classification. To demonstrate the universality of the aspect-location embedding module, we derive a variant model of ALAN (ALAN$_{var}$) without attention support based on it. ALAN$_{var}$ utilizes convolutional neural networks (CNNs Kim 2014) with a gating mechanism to integrate aspect-related sentiment information in context.

ALAN has the following advantages: (I) Our proposed aspect-location embedding module can be applied to most ACSA tasks, and some approaches on ACSA tasks can also use this module as a special embedding module to enhance the initial representations. (II) The attention modeling approach combined with aspect-location embedding can better integrate aspect-specific sentiment features in sentences. (III) The modules in ALAN accomplish their required feature extraction, resulting in an overall low coupling and excellent robustness.

We summarize our main contributions as follows:

- A novel aspect-location embedding module is proposed that dynamically combines aspect categories and contextual information. This is the most prominent contribution of our work.
- An improved attention network based on aspect-location embedding representation is further proposed to aggregate aspect-specific sentiment features in sentences.
- A variant model of ALAN without the support of an attention mechanism is devised, and shows the superiority of the aspect-location embedding method.
- Results on three experimental datasets show that ALAN consistently outperforms other compared baseline models and demonstrate the effectiveness of ALAN and its variant.

The remainder of this article is organized as follows. Section 2 reviews the related work before our method was proposed. Section 3 details the structure and realization process of our model. In Section 4, we present the results and descriptions of all experiments. Section 5 concludes our work and the content of the paper and provides an outlook on future research directions.

2 Related work

2.1 Early research

Most of the early papers use machine learning methods such as Naive Bayes, Maximum Entropy, Logistic Regression, Support Vector Machine (SVM), etc. (Tripathy et al., 2016;

Al-Smadi et al., 2017). These methods focus on traditional sentiment analysis and aspect-term sentiment classification, which cannot be fully applied to ACSA tasks, but have the significance of borrowing. The basic idea is to apply these algorithms to predict the most likely class based on a complex combination of features, but such features usually need to be designed manually. Varghese and Jayasree (2013) combined dependency parsing, co-reference parsing, and SentiWordNet, culminating in SVM as the primary classifier. Singh et al. (2013) proposed aspect-term sentiment classification of movie reviews using different linguistic features and n-gram feature extraction based on SentiWordNet scheme. Karagoz et al. (2019) proposed a framework that focuses on aspect extraction and aspect sentiment word retrieval by using an unsupervised approach and provided a tool to visualize the analysis results. The methods mentioned above have achieved some results. However, due to the explosive growth of data on platforms such as social networks, there are great obstacles to the application of traditional sentiment analysis methods in the era of big data.

2.2 Deep learning methods for ACSA

The rapid development of neural networks and deep learning has driven the development of natural language processing, and a large number of deep learning methods applied to ACSA have been proposed. LSTM and GRU (Chung et al., 2014) based on Recurrent Neural Networks have been widely adopted for various sentiment classification tasks. Tang et al. (2016) proposed TD-LSTM and TC-LSTM based on the connection between the target words and their context. TD-LSTM uses two LSTMs to model the sentences before and after the target words respectively, and finally fuses the extracted features to determine the sentiment polarity of the text. In order to better utilize the relationship between the target words and the entire text, TC-LSTM explicitly links the target words to each word in the text based on TD-LSTM as the modeling embedding layer. Although the correlation between the target words and the context is taken into account, such a simple linkage is not sufficient to maximize the internal association.

After that, it was inspired by the success of the attention mechanism in machine translation and the Memory Network's ability to optimize machine reading comprehension (Hermann et al., 2015). Wang et al. (2016) proposed an Attention-based LSTM. This strategy incorporates aspect word embedding to compute attention weights, thereby forcing the model to pay attention to the important part of the text, which has a certain meaning. However, the model only considers aspect content when calculating contextual weights, and the aspect embedding is the same as before. Ma et al. (2017) noticed that the evaluation object and the context representations can be modeled separately and proposed IAN. They utilized an interactive method to calculate the attention weights of the two, and the learned features can be spliced together as sentiment representations.

CNNs have also been proven to work in the field of NLP (Kim, 2014; Ramaswamy & Chinnappan, 2022). Convolution operation can also capture semantically rich text features. TextCNN, proposed by Yoon Kim in (Kim, 2014), applied CNNs to text classification tasks. The core idea is to capture local features. For text, local features are sliding windows consisting of several words, similar to N-grams (Mikolov et al., 2013). Xue and Li (2018) proposed a model based on CNNs and Gating Mechanism. A new gate unit can control the sentiment features of the output with a given aspect-category. Since convolutional layers are not time-dependent, it is possible that the computations during training can be parallelized, thus reducing the time cost. It is the lack of temporal dependency, however, that ignores inter-textual word order features.

With the rise of graph networks, there are also some studies advocating the construction of relational graphs for ACSA tasks. Liang et al. (2021) proposed an aspect-aware graph convolutional network (AAGCN). They design a beta distribution guided aspect-aware algorithm to compute the relational weights between the aspect and the external affective knowledge, and the obtained results are transferred to the syntactic dependency tree of the original sentence. In this way, GCN networks are constructed for aspect category sentiment classification. Subsequently, they investigated a new few-shot aspect category sentiment analysis task in the paper (Liang et al., 2022), and proposed a meta-learning framework in combination with previous aspect-aware information.

2.3 Pre-trained models (PTMs) for ACSA

In recent years, a great deal of research has shown that PTMs based on a large corpus can learn general language representations. This is a result of transfer training, which allows the model to have some experience from the beginning, rather than starting from scratch. There have been several international NLP tasks that can confirm the advantages of PTMs. PTMs have gone through about two stages. The first stage is to train the representation of a single word, so that words with similar semantics or the same category have a certain connection in the vector space, but they are context-independent, such as Word2Vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014). There is a big breakthrough in the second stage. In Kenton and Toutanova (2019), BERT was proposed, which intercepts the encoder of Transformer (Vaswani et al., 2017). The full name of BERT is Bidirectional Encoder Representation from Transformer. It is a language model trained by Google in an unsupervised manner on a large unlabeled corpus. It is able to learn contextually relevant word vectors and obtain a better representation of the text before processing downstream tasks. Xu et al. (2019) viewed ACSA as a new task and they called it Review Reading Comprehension. In fact, a post-training approach to BERT networks was explored, considering aspects and texts as two sentences connected by a special character. Gao et al. (2019) designed a target-dependent model based on BERT (TD-BERT) to locate the output at the target term and an optional sentence with a built-in target. Although it produced good results in context-aware representation, the embedding representation of aspects was not incorporated in the model. Dai et al. (2021) investigated whether PTMs contain sufficient syntactic information for aspect-level sentiment analysis. After experimentally comparing the induced trees from PTMs and the dependency parsing trees, their proposed RoBERTa-MLP demonstrates that PTMs implicitly encompass task-oriented grammatical information. Liu et al. (2021) considered a more direct approach of transforming the ACSA task into a natural language generation task by utilizing the pre-trained language model BART. They designed templated natural language sentences to represent the output, and the last word of the template sentences was used as the basis for determining sentiment polarity.

Recently, some researchers have considered ACSA and rating prediction (RP) as two highly related tasks and combined information from both tasks to construct models. Bu et al. (2021) collected a new dataset of Chinese reviews (called ASAP) containing the required labeled information for both ACSA and RP tasks. They also designed a joint model based on BERT to enhance the accuracy on both tasks. Fei et al. (2022) followed up their research by taking inspiration from human intuition and proposed a from-fine-to-coarse reasoning framework to obtain better performance on the joint task.

3 ALAN model

3.1 Problem definition

ACSA task is to predict the sentiment polarity of a given aspect-category of a piece of text. The input to this problem can be regarded as a tuple (A, X), consisting of an aspect-category and a contiguous segment of text. The text $X = \{x_1, x_2, x_3, ..., x_n\}$ consists of n words, and the aspect-category $A = \{a_1, a_2, a_3, ..., a_m\}$ contains m words. The output of ACSA is the sentiment label $y \in \{1, 2, ..., K\}$ of the given aspect-category, where K stands for the set of sentiment polarity. For ACSA, the number of aspect categories A is always finite and each A has only a few words. In contrast, the text X is arbitrary and generally m is less than n. The scope of y can also be further refined according to the specific task requirements. The same text may have multiple aspect-categories, and aspect-category words may appear directly in the text or may not appear at all.

3.2 Overview of ALAN

The overall architecture of ALAN is shown in Fig. 1. Our model ALAN consists of four modules, which are the semantic representation module, the aspect-location embedding module, the aspect-location attention learning module and the classifier module. In ALAN, the input tuples (A, X) of the ACSA task in the above problem definition first enter the semantic representation module, then are respectively mapped to semantic embedding representations. After that, the aspect category representation and the sentence representation are dynamically fused into aspect-specific embedding representations by the proposed aspect-location embedding module. The original sentence representations pass through the LSTM layer in the aspect-location attention module, and enter the attention layer together with the aspect-specific embedding representations to generate aspect-specific sentiment features through the aspect-location attention mechanism. The final obtained features are

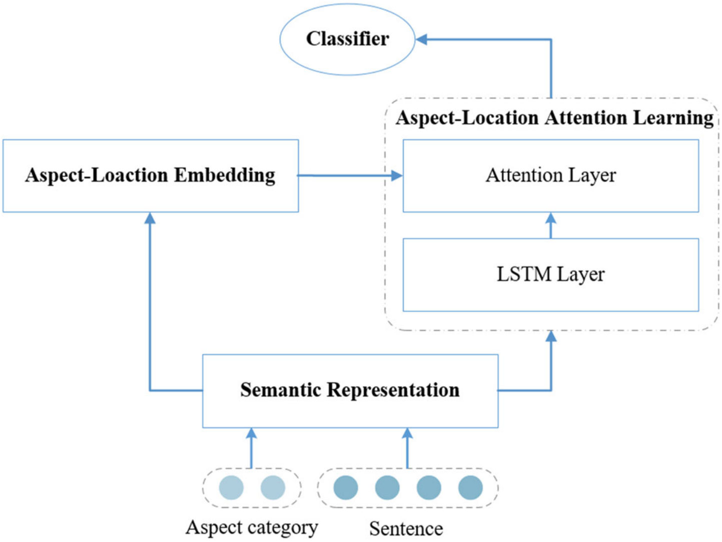

Fig. 1 The overall architecture of ALAN

fed into the classifier module for aspect category sentiment classification. These four modules and their combination methods are described in detail in the following.

3.3 Semantic representation

The main work in this part is to extract the semantic representations of the initial text and aspect-category words. We use the results of the pre-trained model BERT as the basis for our entire model architecture to obtain the embedding representations of the text. BERT has its tokenizer WordPiece, which can convert the text X into a sequence of tokens. After that, the token sequence is expanded into a high-dimensional embedding representation

$$E_x = [e_c, e_1, e_2, ..., e_n, e_s] \tag{1}$$

through the embedding layer. $E_x \in \mathbb{R}^{d\times(n+2)}$ where d is the dimension of the embedding layer of BERT, n is the length of the original text sequence. e_c represents the embedding vector of the first token [CLS], and e_s represents the embedding vector of the last token [SEP]. After the multi-layer encoding of BERT, we take all the hidden states of the last layer as the initial embedding matrix

$$T = [t_c, t_1, t_2, ..., t_n, t_s] \tag{2}$$

where $t_i \in \mathbb{R}^d$ is the final features of each token and it can be seen that the input and output dimensions of BERT are consistent. We also need to convert the aspect category words into an initial embedding matrix T_a and average pool them to obtain the embedding vector $v_a \in \mathbb{R}^d$ of the aspect category.

$$T_a = [t^a_c, t^a_1, t^a_2, ..., t^a_m, t^a_s] \tag{3}$$

$$v_a = \tfrac{1}{m+2}\left(t^a_c + t^a_s + \sum_{i=1}^{m} t^a_i\right) \tag{4}$$

3.4 Aspect-location embedding

This module is used to mine the aspect-category for location information in the text. Figure 2 demonstrates the structure of the aspect-location embedding module. The final text embedding matrix T and the aspect-category embedding v_a obtained from the text semantic representation module are used as the input unit of this part. Multiple aspect-categories may be contained in the same text, but the standard BERT cannot express the different semantic features. Directly predicting the specific sentiment classification of different aspect-categories with this approach does not show any difference and it degrades to the sentiment classification of the whole text. In order to alter the original semantic representation according to the given aspect-category, we propose a method to match location information and design a specialized aspect-location embedding function to construct location features. Specifically, the aspect-category vector v_a and each vector in the vector matrix T are calculated as a similarity score and the index $s_{\max}$ of the vector with the maximum similarity is taken. Then the embedding weight r_i of each vector in the matrix T is calculated by

$$s_{\max} = argmax(v_a^T T) \tag{5}$$

$$r_i = \exp\left(-\tfrac{(i-s_{\max})^2}{2\sigma^2}\right) \tag{6}$$

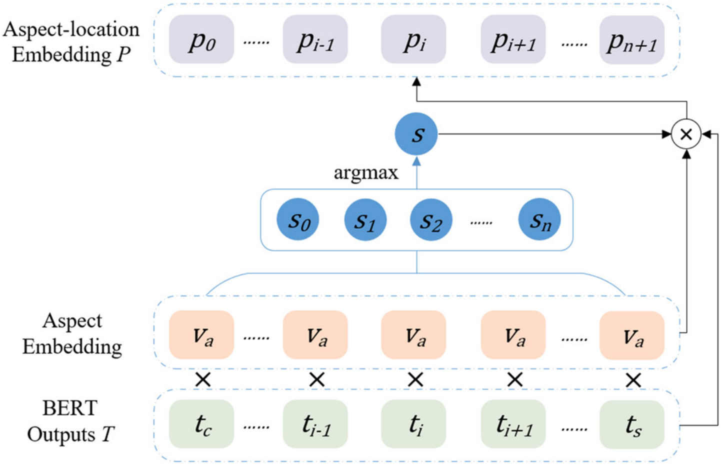
Aspect-location Embedding P
p_0 ...... p_{i-1} p_i p_{i+1} ...... p_{n+1}
s
argmax
s_0 s_1 s_2 ...... s_n
Aspect Embedding
V_a ...... V_a V_a V_a ...... V_a
BERT Outputs T
t_c ...... t_{i-1} t_i t_{i+1} ...... t_s

Fig. 2 The detailed structure of aspect-location embedding module

where v_a^T is the transpose of matrix v_a, $i \in \{0, 1, 2, ..., n\}$ is the index of each word in the input text. $\sigma \in \mathbb{R}$ is the location embedding rate and an adjustable hyperparameter. The reason for this design is that the similarity score can be employed to locate the approximate distribution of aspect-category in the text, and secondly, the weights calculated by our designed aspect-location function can retain more significant features near the aspect-category and the text features farther apart will be weakened. Thereby, an effect of dynamic embedding of aspect-category information into text is achieved.

We further combine text embedding and aspect-category embedding to learn aspect-location representation P, which can make location features more suitable for aspect-category. Mathematically, we compute P as

$$t_i^* = r_i \times t_i \tag{7}$$

$$v_a^i = (1 - r_i) \times v_a \tag{8}$$

$$p_i = t_i^* + v_a^i \tag{9}$$

$$P = [p_0, p_1, p_2, ..., p_n, p_{n+1}] \tag{10}$$

where $p_i \in \mathbb{R}^d$ denotes the ith vector in the aspect-location representation P.

3.5 Aspect-location attention learning

The function of the module is mainly to learn critical information in the text exploiting the attention mechanism. Figure 3 is a schematic diagram of the aspect-location attention learning module. The input units of the module are the text embedding matrix T and the aspect-location embedding P. The standard LSTM is unable to focus on capturing the important semantic features related to aspect-category in the text. To break through this limitation, we design an attention mechanism based on aspect-location embedding P to

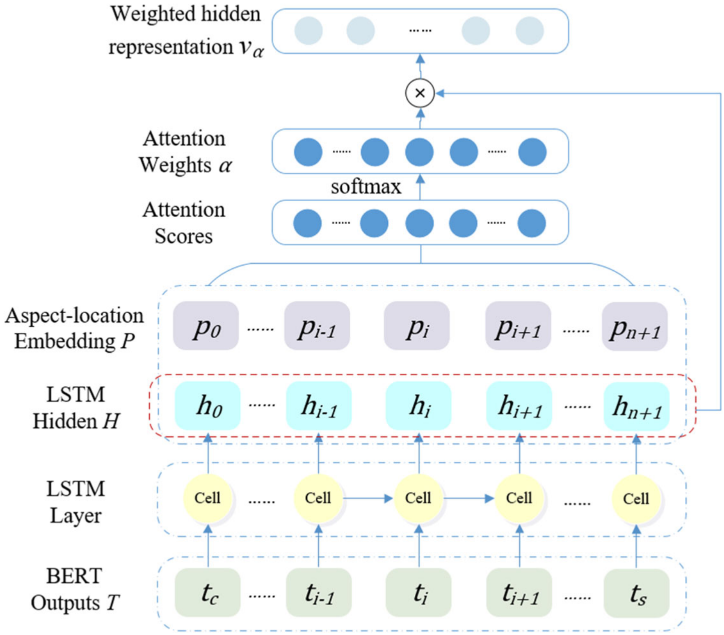
Weighted hidden representation v_α
Attention Weights α
softmax
Attention Scores
Aspect-location Embedding P
p_0 ...... p_{i-1} p_i p_{i+1} ...... p_{n+1}
LSTM Hidden H
h_0 ...... h_{i-1} h_i h_{i+1} ...... h_{n+1}
LSTM Layer
Cell ...... Cell Cell Cell ...... Cell
BERT Outputs T
t_c ...... t_{i-1} t_i t_{i+1} ...... t_s

Fig. 3 The detailed structure of aspect-location attention learning module

improve the LSTM. We put the text embedding matrix T into a LSTM to obtain a sequence of hidden layer vector matrix H consisting of each hidden vector h_i by

$$
\begin{aligned}
f_i &= sigmoid(W_f \cdot [h_{i-1}, n_i] + b_f) && (11)\\
g_i &= sigmoid(W_g \cdot [h_{i-1}, n_i] + b_g) && (12)\\
\tilde{c}_i &= tanh(W_c \cdot [h_{i-1}, n_i] + b_c) && (13)\\
c_i &= g_i \odot \tilde{c}_i + f_i \odot c_{i-1} && (14)\\
o_i &= sigmoid(W_o \cdot [h_{i-1}, n_i] + b_o) && (15)\\
h_i &= o_i \odot tanh(c_i) && (16)\\
H &= [h_0, h_1, h_2, ..., h_{n+1}] && (17)
\end{aligned}
$$

where W_f, W_g, W_c and W_o represent weight matrices while b_f, b_g, b_c, and b_o denote biases.

After obtaining the hidden vector sequence H, we calculate the correlation between each hidden state and the aspect-location embedding matrix P. The aspect-location attention mechanism will generate an attention-weight vector $\alpha \in \mathbb{R}^{n+2}$ and a weighted hidden representation $v_\alpha \in \mathbb{R}^d$.

$$
\begin{aligned}
M &= tanh([W_h H; W_p P]) && (18)\\
\alpha &= softmax(w_\alpha^T M) && (19)\\
v_\alpha &= H\alpha^T && (20)
\end{aligned}
$$

where $M \in \mathbb{R}^{2d\times(n+2)}$, $W_h \in \mathbb{R}^{d\times d}$, $W_p \in \mathbb{R}^{d\times d}$ and $w_\alpha \in \mathbb{R}^{2d}$ are projection parameters.

3.6 Classifier

The basic approach of ALAN is to directly apply aspect-location representation v_α for sentiment classification. A linear layer is added to compress v_α to a length equal to the number of sentiment polarities. We convert to a conditional probability distribution y_α by

$$y_\alpha = softmax(W_\alpha v_\alpha + b_\alpha) \tag{21}$$

where W_α and b_α are projection parameters of the linear layer. The sentiment polarity of the values in the conditional probability distribution y_α is treated as the final sentiment classification prediction.

3.7 The variant of ALAN

ALAN$_{var}$ is constructed on the basis of the aspect-location embedding module, as shown in Fig. 4. It transforms aspect-location embedding representations into n-gram features in sentences, and then maximally pools these features to obtain sentiment representations for the corresponding aspect category.

The aspect-location embedding representations are integrated with multiple (two in our experiments) convolutional networks with different convolutional kernel sizes, where different activation functions are utilized to control the range of the output. The results of the convolution are successively subjected to pooling and concatenating operations to learn different feature representations. We compute two convolutional representations $c_i^t \in \mathbb{R}^d$ and $c_i^r \in \mathbb{R}^d$ by

$$c\ _i^t = \tanh(P_{(i:i+k_1)} * W_t + b_t) \tag{22}$$

$$c\ _i^r = \mathrm{relu}(P_{(i:i+k_2)} * W_r + b_r) \tag{23}$$

where $*$ represents the convolution operation, b_t, b_r are the bias, and k_1, k_2 are the size of the convolution kernel. The activation functions are $tanh$ and $relu$. Different convolution kernel sizes are set in the above two equations, allowing the obtained convolutional features to be more representative and present more specific aspect-location information. According to the characteristics of the two activation functions, the $tanh$ function can generate features that conform to the semantics of the text more consistently, while $relu$ can additionally accept the changing features generated by the combination of text and aspect-category. The two convolutional sequences are max pooled separately, and the resulting vectors are concatenated to obtain the final aspect-location representation vector $v_\beta \in \mathbb{R}^{2\times d}$.

$$v_\beta = c_m^t \| c_m^r \tag{24}$$

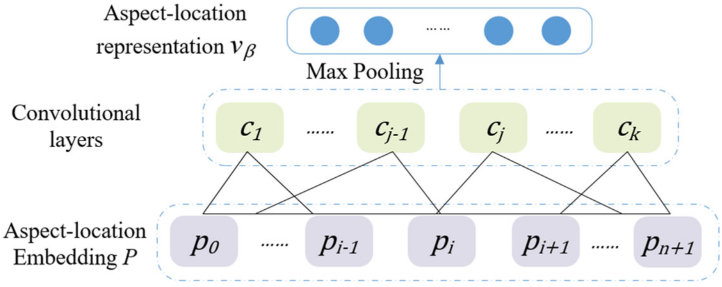

Fig. 4 The structure of ALAN$_{var}$

where $\|$ represents a concatenation operation, c_m^t and c_m^r are the vectors of the two convolutional sequences after max pooling. Finally, v_β is fed into the classifier module to obtain the conditional probability distribution y_β of the sentiment polarities.

$$y_\beta = softmax(W_\beta v_\beta + b_\beta) \tag{25}$$

where W_β denotes weight matrices while b_β denotes biases.

3.8 Model efficiency analysis

We assume that the length of a sentence is n and the dimension of the embedding vector is d. The main time overhead of the semantic representation module lies in the multi-layer self-attention mechanism computed in BERT, so the time complexity of the process is $O(n^2d)$. The aspect-location embedding module mainly includes the computation of maximum similarity and aspect-location embedding representations, and their time complexity is $O(n^2d)$ and $O(nd)$, respectively. The aspect-location attention learning module goes through LSTM and aspect-location attention mechanism successively, where the time complexity of LSTM is $O(nd^2)$ and the time complexity of aspect-location attention mechanism is $O(n^2d)$. Generally speaking, d is too large to be ignored. Therefore, the time complexity of ALAN is $O(n^2d)$ without restricting the sentence length.

ALAN$_{var}$ also contains the semantic representation module and the aspect-location embedding module, so the time overhead of this part is consistent. The difference is that ALAN$_{var}$ is followed by the use of convolutional computation. Assuming that the size of the convolution kernel is k (the other size in the text is d by default), the required time overhead is $O(knd^2)$. Since the overhead of the convolution operation is much smaller than the first two modules, the time complexity of ALAN$_{var}$ is just as $O(n^2d)$.

From the above theoretical analysis, the semantic representation module and the aspect-location embedding module account for the major time overhead. Although ALAN and ALAN$_{var}$ end up with the same time complexity, the computation of LSTM relies on the results of the previous time step for each time step, while CNNs can be computed in parallel. Therefore, in practice, the model efficiency of ALAN$_{var}$ is higher than that of ALAN.

4 Experiments

4.1 Datasets and experiment preparation

We have verified the effect of ALAN and ALAN$_{var}$ on three publicly available social review datasets, including two in English and one in Chinese.

The English datasets are the review data on the restaurant field in SemEval-2014 (Manandhar, 2014), SemEval-2015 (Pontiki et al., 2015) and SemEval-2016 (Pontiki et al., 2016). The SemEval-14 dataset has 5 aspect-categories ("food", "anecdotes/miscellaneous", "service", "ambience", "price") as target objects for sentiment classification, and the SemEval-15, SemEval-16 datasets include 12 aspect-categories. Since the SemEval-16 dataset is extended on the SemEval-15 dataset, a large portion of their training data is consistent, so we merge their training and testing datasets separately. The data labeled "conflict" in the original datasets are excluded. For example, the sentence "not a large place, but it's cute and cozy" has a sentiment label of "conflict" for the "ambiance" aspect-category. We only keep data with sentiment labels as "positive", "negative" and "neutral". Since the amount of "neutral" data in the training set of SemEval-15&16 is too small, the "neutral" category

Table 1 Dataset statistics

Dataset	Positive	Negative	Neutral	Total
SemEval-2014 (train)	2177	839	500	3516
SemEval-2014 (test)	657	222	94	973
SemEval-15&16 (train)	1505	713	525	2743
SemEval-15&16 (test)	923	526	87	1536
AC-SemVal-2018 (train)	7409	4438	2553	14400
AC-SemVal-2018 (test)	4359	529	1576	8464
AC-SemTra-2018	82570	22840	36670	142080

cannot be distinguished during the training process, which has a great negative impact on all models. Therefore, we add four copies of the original "neutral" data to the training set, which is a method of data augmentation.

The Chinese dataset is the dataset of the "Fine-grained user comment sentiment analysis" track in "Global AI Challenger 2018". It contains comment data on a social platform. The dataset is divided into four parts: training, validation, test A and test B. The evaluation objects in the dataset are divided into two levels according to different granularities. The first level is the coarse-grained evaluation object, such as "service" and "location" involved in the review text; the second level is the fine-grained emotion object, such as "waiter's attitude" and "waiting time" in the "service" category. There are four sentiment polarities for every fine-grained element: positive, neutral, negative, and unmentioned, which are labeled as 1, 0,− 1 and − 2. We simplify the second level by setting the sentiment polarity of each aspect-category in the first level according to the strategy of "majority voting" on fine-grained sentiment labels. The samples for the "unmentioned" labels are discarded. We divide the data of the original validation set into training data and test data after processing, and the dataset is called "AC-SemVal-2018". The statistics of these datasets are shown in Table 1, and the distribution of aspect-categories in each dataset is shown in Fig. 5.

4.2 Compared methods and experimental settings

We select some of the proposed and powerful baseline methods, conduct experimental comparisons, and evaluate our models. The following is a description of the comparison models:

- **LSTM** (Hochreiter et al., 1997): Since the standard LSTM cannot combine any aspect-level information, the same text is given different aspect-categories and the final predicted sentiment polarity is the same.
- **TextCNN** (Kim, 2014): TextCNN uses three CNNs with different convolutional kernel sizes to convolve text features and stitch the pooled results together as the final representation.
- **ATAE-LSTM** (Wang et al., 2016): The model combines aspect-category word vectors and LSTM-encoded hidden state sequences to learn attention weights, and weights all hidden vectors as aspect-level sentiment classification representations.
- **IAN** (Ma et al., 2017): IAN models the aspect-category and the input text separately and designs an interactive attention mechanism based on the two LSTMs.
- **GCAE** (Xue & Li, 2018): This method proposes an efficient model based on CNNs and Gating Mechanisms. The Tanh-ReLU gating unit retrieves the important information in the text based on the given aspect-category.

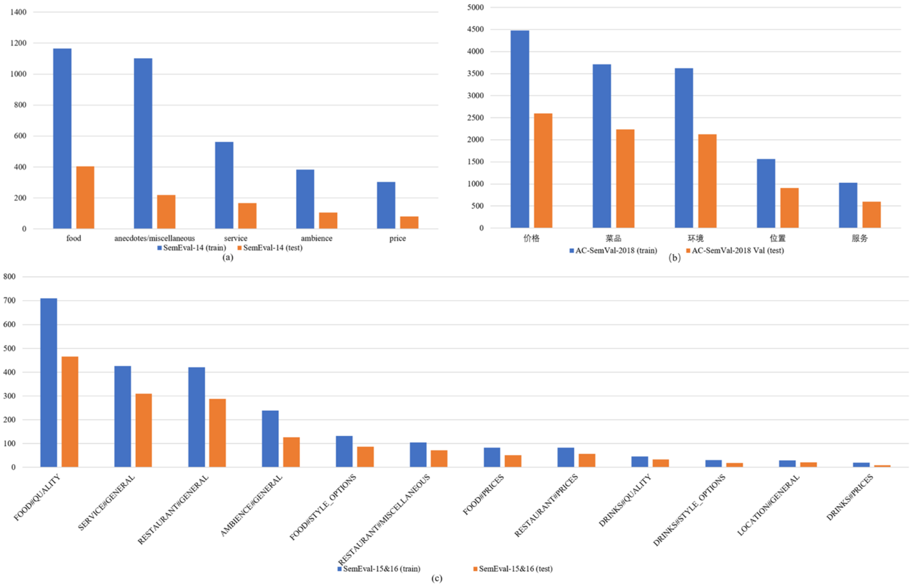

Fig. 5 The distribution of aspect-categories: (a) SemEval-2014; (b) AC-SemVal-2018; (c) SemEval-15&16

- **BERT-PT** (Xu et al., 2019): BERT-PT combines the aspect words and the text as two sentences and inputs them into the BERT model for training, and the obtained [CLS] vector is used as the final representation of the sentiment polarity.
- **TD-BERT-QA-CON** (Gao et al., 2019): This method is a variant of TD-BERT and focuses on fine-tuning the output of the BERT model: pooling the target words and splicing [CLS] vectors as the final representation.
- **BERT-MLP** (Dai et al., 2021): A simple and effective baseline model (RoBERTa-MLP/BERT-MLP) was proposed in Dai et al. (2021). It takes the output of the target words in RoBERTa/BERT and adds a maximum pooling layer and a multi-layer perceptron (MLP) to perform sentiment classification.
- **AAGCN-BERT** (Liang et al., 2021): The model utilizes the beta distribution to calculate the relational weights of aspect category words with external sentiment knowledge and constructs graph networks on the syntactic dependency trees of the sentences.
- **BART generation** (Liu et al., 2021): The model uses natural language generation to design template sentences to represent the output, and takes the features of the last word of the generated sentences as the basis for determining sentiment polarity.

The word vector initialization tool Word2Vec,[1] used for the English datasets (SemEval-14/SemEval-15&16), contains 300 million common words and word vectors, trained by Google using a large amount of computational power based on the huge corpus of Google News, with 300 dimensions per word vector. The words of the vocabularies in Word2Vec are randomly initialized to a uniform distribution U (–0.5, 0.5). We only used the 200,000 most frequently used words, treating the others as unfamiliar and initializing them randomly. For the Chinese dataset (AC-SemVal-2018), the Word2Vec model[2] produced by Li et al.

[1] https://s3.amazonaws.com/dl4j-distribution/GoogleNews-vectors-negative300.bin.gz
[2] https://github.com/Embedding/Chinese-Word-Vectors

Table 2 The main hyperparameter settings

Hyperparameter	SemEval-14	SemEval-15&16	AC-SemVal-2018
Epoch	8	8	16
Batch size	16	16	16
Max length	64	64	256
Learning rate	2e–5	2e–5	2e–5
Optimizer	Adam	Adam	Adam

(2018) and Qiu et al. (2018) is used, which is trained based on a large number of Weibo corpora. The BERT models used in the experiments, for the English and Chinese datasets, are bert-base-uncased[3] and bert-base-chinese,[4] respectively.

Since TD-BERT-QA-CON and BERT-MLP analyze the sentiment of target words in the text, while the object of study in this paper is the aspect-category, which does not necessarily appear directly in the text. Therefore, we take the approach of matching target words. We calculate the word with the highest similarity to aspect-category in the text and take one word from its above and one word from its below as the target words of the given aspect-category.

All the compared models are constructed in the Tensorflow (Abadi et al., 2016) framework and trained on a single NVIDIA TITAN RTX GPU device (24GB RAM). In all experiments, dataset preprocessing is performed. In the English datasets, all uppercase letters are converted to lowercase. To remove deactivated words, we process the Chinese dataset with a list of deactivated words collected on the web. And we remove words related to time, numbers and symbols as they are not relevant to the sentiment classification task. Regarding the text length, if the maximum text length exceeds the set value, it is truncated at the end of the text and preceded by zero if it is insufficient. For the parameters involved in the baseline methods, the values from the original paper are applied or several experiments are performed to select the best values. With BERT as the model for the embedding layer, the parameters are set according to the recommended parameters in the BERT source code.[5] In our method, the embedding size of all word vectors is 768 and other hyperparameters including Epoch, Batch size, Max sequence length, Learning rate, and Optimizer are approved in Table 2. Since the datasets used in the experiments do not provide a specified validation set, we randomly selected 10% of the data samples from the training set as the validation set to adjust the above hyperparameters. We also adopted this approach in the compared experiments to ensure the fairness of the experiments.

4.3 Results analysis of aspect-category sentiment classification

In order to validate the performance effectiveness of ALAN and ALAN$_{var}$, we use the overall classification accuracy (acc) and macro F1-score (F1) as performance evaluation criteria when the training datasets are completely consistent. Table 3 show the performance of ALAN and ALAN$_{var}$ compared with other baseline models on the three datasets.

In general, the standard LSTM and TextCNN perform poorly, especially in terms of macro F1-score, where they lag behind other methods by a large margin. Although their

[3] https://huggingface.co/bert-base-uncased

[4] https://huggingface.co/bert-base-chinese

[5] https://github.com/google-research/bert

Table 3 The performance of ALAN compared with other baseline models. The bold emphasis indicates the maximum value of the performance comparison in its column

	Method	SemEval-14		SemEval-15&16		AC-SemVal-2018	
		Acc	F1	Acc	F1	Acc	F1
Baselines	LSTM	79.20	61.13	77.15	62.70	51.06	45.15
	TextCNN	79.30	60.84	80.79	70.43	53.86	45.36
	ATAE-LSTM	77.46	63.71	76.17	64.33	61.79	46.16
	IAN	77.46	61.27	74.28	60.38	64.04	54.85
	GCAE	80.33	66.71	81.84	72.10	61.98	52.45
	BERT-PT	87.50	78.82	89.19	82.54	70.42	65.10
	TD-BERT-QA-CON	87.09	76.86	88.15	80.38	63.09	54.53
	BERT-MLP	87.70	78.39	88.41	80.95	62.81	56.20
	AAGCN-BERT	89.00	81.81	**90.62**	79.19	–	–
	BART generation	89.45	82.18	89.45	81.46	70.86	66.20
Ablation study	word2vec-ALAN$_{var}$	80.84	68.81	83.79	75.52	61.12	53.56
	word2vec-ALAN	82.17	71.08	80.53	69.15	64.97	55.54
	ALAN$_{var/AE}$	88.01	79.88	87.37	78.80	60.62	54.39
	ALAN$_{/AE}$	88.52	80.26	87.96	80.52	61.97	55.86
	ALAN$_{/Atten}$	88.63	81.78	87.43	80.16	62.12	54.31
Proposed models	ALAN$_{var}$	89.45	**82.56**	89.78	**82.71**	**71.57**	**67.09**
	ALAN	**89.55**	82.22	88.54	80.93	71.38	66.23

overall classification accuracy is higher than ATAE-LSTM and IAN in the English datasets, this is only because the training results of ordinary neural networks will be more skewed towards the classes with more identical labels in the training samples. The underlying reason for its poor performance is that the aspect-category information is not considered, such that each word is equal in the neural network, which will affect the sentiment classification effect of different aspect-categories. From the experimental results on Chinese dataset, the performance of standard LSTM and TextCNN is similarly far inferior to that of the model considering aspect-category. Therefore, these two methods may be more suitable for common text classification tasks. GCAE outperforms ATAE-LSTM and IAN in the English datasets and slightly underperforms than IAN in the Chinese dataset because the attention mechanism adopted by ATAE-LSTM and IAN can combine the supervision role of aspect-category information to effectively obtain important contextual information. Due to its special gating mechanism, GCAE can obtain richer features in combination with CNNs. The methods (BERT-PT, TD-BERT-QA-CON, BERT-MLP) fine-tuned based on the BERT model consistently surpass previous methods on all datasets. Although the overall classification accuracy of TD-BERT-QA-CON and BERT-MLP is slightly inferior to IAN on the Chinese dataset, their macro F1-scores are also stable and better than ATAE-LSTM, IAN and GCAE. Benefiting from its exploitation of external knowledge and rich graph structure, AAGCN-BERT has slightly higher accuracy than ALAN and ALAN$_{var}$ on SemEval-15&16. From the experimental results of BART generation, although it is slightly inferior to ALAN and ALAN$_{var}$, it still outperforms all the compared BERT-based methods.

This is mainly due to the differences in the pre-trained corpus and the number of parameters, where BART is much superior to BERT.

Our ALAN and ALAN$_{var}$ further deepen the importance of aspect-category objects and their related contextual information by proposing a combination of aspect-category location distribution features and special location attention mechanisms. From the experimental results, our proposed ALAN method consistently outperforms all contrasting methods (accuracy improved by 0.1% on SemEval-14., 0.71% on AC-SemVal-2018. macro-F1 improved by 0.38% on SemEval-14., 0.17% on SemEval-15&16., 0.89% on AC-SemVal-2018). State-of-the-art results are achieved on the three datasets involved in the experiments in terms of two performance metrics (Acc and F1).

To test whether the performance between methods is statistically significantly different, we applied the non-parametric McNemar's test (Dietterich, 1998).This test is well suited for our purposes because it does not require a normal distribution of the data and has also been used in related studies (Chen et al., 2019). In order to make a comparison between method A and method B by the McNemar's test, we need to count the number of samples that are correctly classified by A instead of B (denoted as n_{10}) and the number of samples that are correctly classified by B instead of A (denoted as n_{01}). Then we can compute the statistic

$$\chi^2 = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}} \tag{26}$$

which is distributed as χ^2 with 1 degree of freedom. Performance is considered to be statistically significantly different only if the p-value of the computed statistic is below a pre-specified significance level.

The results of the statistics are shown in Tables 4 and 5, and we specified a significance level of 5%. From the data in the tables, the performance of our proposed ALAN and ALAN$_{var}$ on the three datasets is mostly superior to that of the baseline methods, and only individual methods cannot present significant differences. It was previously noticed that on SemEval-15&16, the accuracy of AAGCN-BERT was higher than slightly ALAN$_{var}$, while macro-F1 was much lower than ALAN$_{var}$, so it was difficult to distinguish their performance. However, from the results of McNemar's test, the p-value of ALAN$_{var}$ vs.

Table 4 McNemar's statistics between the results of ALAN$_{var}$ and other methods on each dataset

Method	SemEval-14		SemEval-15&16		AC-SemVal-2018	
	χ^2	p	χ^2	p	χ^2	p
LSTM	57.309⋆	0.000	118.534⋆	0.000	963.628⋆	0.000
TextCNN	55.249⋆	0.000	71.969⋆	0.000	835.706⋆	0.000
ATAE-LSTM	72.901⋆	0.000	129.063⋆	0.000	350.216⋆	0.000
IAN	71.405⋆	0.000	160.507⋆	0.000	255.314⋆	0.000
GCAE	49.091⋆	0.000	64.574⋆	0.000	368.057⋆	0.000
BERT-PT	4.938⋆	0.026	0.480	0.488	22.830⋆	0.000
TD-BERT-QA-CON	7.680⋆	0.006	18.618⋆	0.000	364.618⋆	0.000
BERT-MLP	4.696⋆	0.030	8.113⋆	0.004	345.211⋆	0.000
AAGCN-BERT	0.404	0.525	8.028⋆	0.005	–	–
BART generation	0.012	0.914	0.085	0.771	11.653⋆	0.001

⋆ means the result is significant at the 5% level

Table 5 McNemar's statistics between the results of ALAN and other methods on each dataset

Method	SemEval-14		SemEval-15&16		AC-SemVal-2018	
	χ^2	p	χ^2	p	χ^2	p
LSTM	59.172⋆	0.000	95.508⋆	0.000	882.049⋆	0.000
TextCNN	60.500⋆	0.000	54.179⋆	0.000	761.274⋆	0.000
ATAE-LSTM	77.778⋆	0.000	106.313⋆	0.000	333.638⋆	0.000
IAN	79.587⋆	0.000	136.172⋆	0.000	223.017⋆	0.000
GCAE	52.112⋆	0.000	45.039⋆	0.000	323.218⋆	0.000
BERT-PT	5.470⋆	0.019	0.736	0.391	9.339⋆	0.002
TD-BERT-QA-CON	8.817⋆	0.003	2.761	0.097	292.638⋆	0.000
BERT-MLP	4.516⋆	0.034	0.456	0.500	277.095⋆	0.000
AAGCN-BERT	0.329	0.566	2.913	0.088	–	–
BART generation	0.011	0.915	1.432	0.231	3.556	0.059

⋆ means the result is significant at the 5% level

AAGCN-BERT is significant at 5% level. Thus, it can be verified that ALAN and ALAN$_{var}$ have superior performance.

4.4 Ablation study

To further investigate how different components of ALAN affect the performance of ACSA, we evaluate the performance of the ablated ALAN model. The ablation model excludes BERT semantic representations (word2vec-ALAN$_{var}$, word2vec-ALAN), aspect-location embedding (ALAN$_{var/AE}$, ALAN$_{/AE}$), and aspect-location attention mechanisms (ALAN$_{/Atten}$), respectively. In word2vec-ALAN$_{var}$ and word2vec-ALAN, we use Word2Vec instead of BERT as the word embedding layer. The results of the comparison of the complete ALAN model with its ablation are shown in Table 3.

As a whole, the performance of ALAN$_{var}$ and ALAN suffers from different degrees of impairment after removing important components, and there are some differences across datasets. We observe that ALAN and ALAN produce some degree of degradation in their effectiveness on all datasets after losing the support of BERT. However, compared to all Word2Vec-based baseline models (LSTM, TextCNN, ATAE-LSTM, IAN and GCAE), the performance of word2vec-ALAN$_{var}$ and word2vec-ALAN is still superior (accuracy improved by 1.84% on SemEval-14., 1.95% on SemEval-15&16., 0.93% on AC-SemVal-2018. macro-F1 improved by 4.37% on SemEval-14., 3.42% on SemEval-15&16., 0.69% on AC-SemVal-2018). Compared with the full model, ALAN$_{var/AE}$, ALAN$_{/AE}$ and ALAN$_{/Atten}$ have small decreases in accuracy and F1 scores, around 1-2%, on the two commonly small datasets of sentiments (SemEval-14, SemEval-15&16). Their performance slips about 10% on the larger Chinese dataset (AC-SemVal-2018), even lower than Word2Vec-based ALAN$_{var}$ and ALAN. It demonstrates the effectiveness of aspect-location embedding and aspect-location attention mechanism, which are indispensable for the proposed model. It can also be seen that BERT can show excellent performance on small-scale datasets with only fine-tuning, but for some larger and more complex datasets, special neural networks need to be designed to adapt.

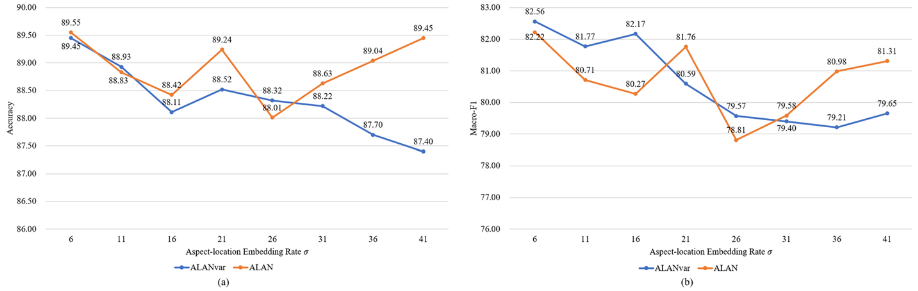

Fig. 6 The performance of different σ in SemEval-14: (a) Accuracy (%); (b) Macro-F1(%)

4.5 Impact of the proposed parameter

For the location embedding rate σ proposed in our method,we recommend to take a wide range in the experiments, which varies according to the length of the text, and then use a large step size to select candidate values within the range and verify the best value. As an example, for the SemEval-14 dataset, we change σ from 6 to 51 in increments of 5. For the Chinese dataset (AC-SemVal-2018), since the length of the Chinese text is much larger than that of the English text, we choose a larger range for the candidate value interval. We changed its value from 16 to 64 in increments of 8.

Figures 6, 7 and 8 show the changes in the performance of our models after adjusting σ on each dataset. We evaluate the model effect with classification accuracy and macro F1-score, train the model after determining the value of σ and take the optimal result for analysis. From the results shown in the line graphs, the aspect embedding rate σ has a more obvious effect on the performance of ALAN$_{var}$ and ALAN. The performance of ALAN$_{var}$ and ALAN peaks around $\sigma = 6$ on SemEval-14 and SemEval-15&16. On AC-SemVal-2018, it peaks around $\sigma = 24$. The trend of the lines shows that ALAN$_{var}$ has a relatively steady decreasing tendency in performance as σ increases. Although the change in the performance of ALAN is not significant, it can still be seen that the effect is better when σ is smaller. This is because as σ increases, the aspect-location embedding features will become smooth until become undifferentiated aspect category embedding. Thus, aspect-specific representations are missing, leading to a decrease in model performance.

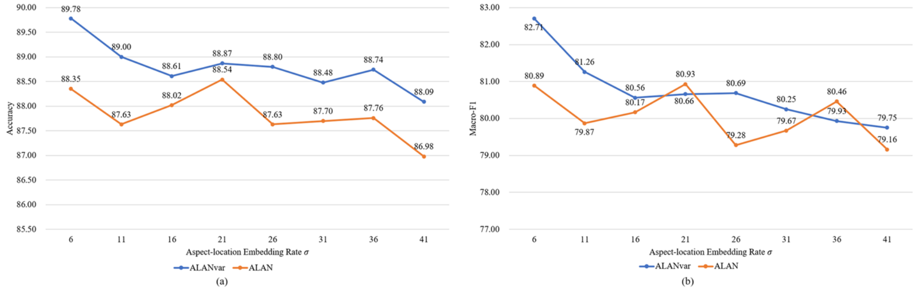

Fig. 7 The performance of different σ in SemEval-15&16: (a) Accuracy (%); (b) Macro-F1(%)

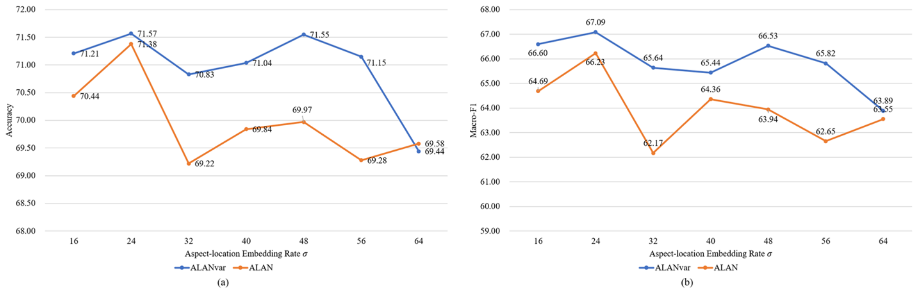

Fig. 8 The performance of different σ in AC-SemVal-2018: (a) Accuracy (%); (b) Macro-F1(%)

4.6 Stability evaluation

The aim of this section is to validate the model trained on a small dataset and analyze whether it can maintain the same performance (defined as stability of predictive performance) when predicting a large amount of data. The experiment also avoids the performance errors of small test sets. For this purpose, we provide a large Chinese test dataset. This dataset is obtained by processing the training dataset of the "Fine-grained user comment sentiment analysis" track in "Global AI Challenger 2018". It is the same source dataset as AC-SemVal-2018, but has no intersection and is noted as AC-SemTra-2018. Its processing method is the same as AC-SemVal-2018, see Section 4.2 for details, and the data distribution is recorded in Table 1. Table 6 shows the results of predicting AC-SemTra-2018 by using the model trained on AC-SemVal-2018. It clearly shows that our models are robust, consistently superior to the baseline methods by a significant margin, and achieve higher overall classification accuracy. We find a slight decrease in macro F1-scores, so we further investigate the classification effect of the three sentiment polarities. Figure 9 shows a comparison of the specific classification effects of our model on AC-SemVal-2018 and AC-SemTra-2018. From Fig. 9, we can see that the ALAN$_{var}$ and ALAN do not change much in predicting the sentiment labels "Positive" and "Negative", but the prediction performance for

Table 6 The performance of our methods and baseline models in AC-SemTra-2018. The bold emphasis indicates the maximum value of the performance comparison in its column

	Method	AC-SemTra-2018	
Baselines	LSTM	48.36	41.82
	TextCNN	57.63	45.07
	ATAE-LSTM	61.84	42.13
	IAN	65.24	53.22
	GCAE	64.16	51.76
	BERT-PT	70.49	62.79
	TD-BERT-QA-CON	64.01	50.06
	BERT-MLP	63.46	52.54
	BART generation	71.36	64.80
Proposed models	ALAN$_{var}$	**72.23**	**65.78**
	ALAN	71.69	64.36

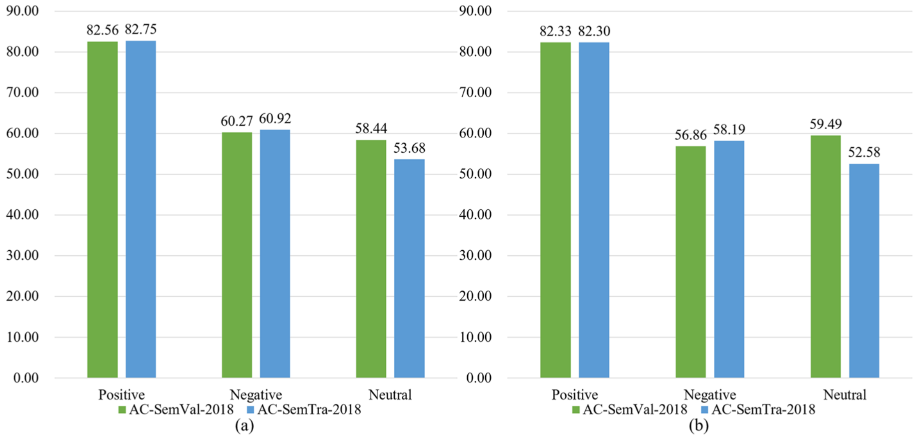

Fig. 9 F1-scores of the three sentiment polarities: (a) ALAN$_{var}$; (b) ALAN

"Neutral" decreases a bit. It can be inferred that the models have some difficulty in capturing the sentiment features in the text when predicting the text with sentiment polarity "Neutral".

4.7 Case study

4.7.1 Qualitative evaluation

We randomly select some qualitative examples from the test data of SemEval-2014 and present the prediction results of the proposed models ALAN$_{var}$ and ALAN and a baseline model BERT-PT in Table 7.

Table 7 Example predictions on SemEval-2014 (test)

Example	BERT-PT	ALAN$_{var}$	ALAN
1.Meal was very expensive for what you get. **[Price]**$_{neg}$	[Price]$_{neg}$(✔)	[Price]$_{neg}$(✔)	[Price]$_{neg}$(✔)
2.The folding chair I was seated at was uncomfortable. **[Ambience]**$_{neg}$	[Ambience]$_{neg}$(✔)	[Ambience]$_{neg}$(✔)	[Ambience]$_{neg}$(✔)
3.As we waited I watched 3 separate groups of diners discuss how disappointed they also were. **[Anecdotes/miscellaneous]**$_{neg}$	[Anecdotes/ miscellaneous]$_{neg}$(✔)	[Anecdotes/ miscellaneous]$_{neg}$(✔)	[Anecdotes/ miscellaneous]$_{neg}$(✔)
4.The staff should be a bit more friendly. **[Service]**$_{neg}$	[Service]$_{pos}$(✗)	[Service]$_{neg}$(✔)	[Service]$_{pos}$(✗)
5.While there's a decent menu, it shouldn't take ten minutes to get your drinks and 45 for a dessert pizza. **[Food]**$_{pos}$ **[Service]**$_{neg}$	[Food]$_{pos}$(✔) [Service]$_{pos}$(✗)	[Food]$_{pos}$(✔) [Service]$_{neg}$(✔)	[Food]$_{pos}$(✔) [Service]$_{neg}$(✔)
6.Once we sailed, the top-notch food and live entertainment sold us on a unforgettable evening. **[Food]**$_{pos}$ **[Ambience]**$_{pos}$	[Food]$_{pos}$(✔) [Ambience]$_{pos}$(✔)	[Food]$_{pos}$(✔) [Ambience]$_{neg}$(✗)	[Food]$_{pos}$(✔) [Ambience]$_{pos}$(✔)
7.Although the restaurant itself is nice, I prefer not to go for the food. **[Food]**$_{neg}$ **[Ambience]**$_{pos}$	[Food]$_{pos}$(✗) [Ambience]$_{pos}$(✔)	[Food]$_{neg}$(✔) [Ambience]$_{neg}$(✗)	[Food]$_{pos}$(✗) [Ambience]$_{pos}$(✔)
8.I found the food to be just as good as its owner, Da Silvano, just much less expensive. **[Food]**$_{pos}$ **[Price]**$_{pos}$ **[Service]**$_{pos}$	[Food]$_{pos}$(✔) [Price]$_{neg}$(✗) [Service]$_{neg}$(✗)	[Food]$_{pos}$(✔) [Price]$_{pos}$(✔) [Service]$_{pos}$(✔)	[Food]$_{pos}$(✔) [Price]$_{pos}$(✔) [Service]$_{pos}$(✔)

Firstly, there are cases where only one aspect category exists in the sentence. There is a high probability of obvious emotions in such sentences (“expensive” in example 1, “uncomfortable” in example 2 and “disappointed” in example 3), and the three models are able to predict their sentiment polarity relatively accurately. However, similar to Example 4, when multiple words “a bit more friendly” are required to jointly judge the sentiment of the aspect category “service”, the performance of BERT-PT and ALAN is not very good. This is mainly because these two models unilaterally judge the sentiment polarity by the sentiment word “friendly”. Furthermore, in examples 5 to 8, the number of aspect categories in the sentence is more than one. Obviously, it is more difficult to correctly predict their respective sentiment polarities. BERT-PT, which does not consider the embedding of aspect categories, performs well only in Example 6, and its correct prediction is due to the explicit sentiment words “top-notch” and “unforgettable” in the context. But for some other examples, such as examples 5, 7 and 8, where the polarities of some aspect categories cannot be directly predicted by their complex contexts, BERT-PT performs much less well than ALAN$_{var}$ and ALAN. In examples 6 and 7, the lack of an attention mechanism weakened the focus on some important sentiment features, which led to the poor performance of ALAN$_{var}$.

Overall, ALAN$_{var}$ and ALAN are better than BERT-PT in these cases, although they also have some misjudgment phenomena. For the above mentioned misclassification cases, two mitigation methods are considered. One is to extend the attention mechanism from a single head to multiple heads, thus stabilizing the attention learning process. The other is to increase the samples similar to the error cases, and some data augmentation can be employed to collect more samples to train the model. These schemes will be continued and validated in our future practice.

4.7.2 Validation of attention

To investigate whether the attention mechanism of ALAN is effective, we propose an intuitive method to detect the matching of aspect-category with relevant words in the text. This is achieved by visualizing the attention distribution of all words in the text by drawing a heat map, the attention distribution ρ is calculated by

$$\rho = softmax(n \times \alpha[1 : n + 1]) \tag{27}$$

where n is the length of the original text sequence. The formula $\alpha[1 : n + 1]$ represents the weight from the first word to the last word in the attention weight vector α.

In this section, we study two exemplary reviews from SemEval-14 as experimental cases. ALAN is applied to both cases and the correct sentiment classification is obtained. In Case 1, Fig. 10 shows two distinct attention distributions for the sentence “*the food is reliable and the price is moderate*” with two given aspect-categories. The shade of the color represents the intensity of attention. In other words, the darker the color is, the more important the word in that part is. For the explicit aspect-category “*food*”, the word “*reliable*” has a red

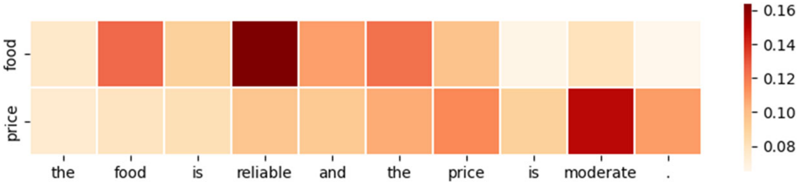

Fig. 10 Case 1, the explicit aspect-categories are “service” and “ambience”

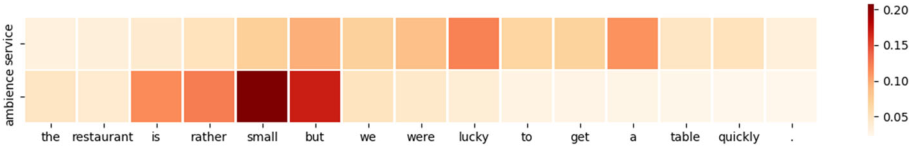

Fig. 11 Case 2, the implicit aspect-categories are "food" and "price"

to black color, and similarly for the aspect category "*price*", the word "*moderate*" has a striking color. Obviously, this is in line with our human judgment on semantic emotion in natural language and it also shows that ALAN can locate emotional information of explicit aspect-categories very well. In Case 2, the given aspect-categories "service" and "ambience" do not appear in direct words in the sentence "*the restaurant is rather small but we were lucky to get a table quickly*". This situation is actually a big difficulty, after all, the machine cannot understand the overall semantics of the sentence as well as a human can. However, in Fig. 11, we find that ALAN focuses most of its attention on the first half of the sentence "*the restaurant is rather small*" according to "ambience", and pays more attention to the second half of the sentence "*but we were lucky to get a table quickly*" under the aspect-category "service". Therefore, to a certain extent, we can see that our model is effective in grasping the overall semantics of the text. In addition, the prepositions and punctuation marks in the text are rarely noticed by ALAN, as can be seen from Figs. 10 and 11. That is consistent with the normal logic of judging sentiment polarity: we do not care about these common words.

As we expected, aspect-category related words and sentiment characteristic words are attended to by ALAN and play a dominant role in correctly judging sentiment polarity. Thus, we conclude that our aspect-location attention mechanism captures important information and is able to model the overall semantics of the text.

5 Conclusion and future work

In this paper, we propose a memory neural network incorporating aspect-location embedding and aspect-location attention mechanism for the ACSA task. Aspect-location embedding can be combined with the idea of attention to form a new attention model (ALAN), which pays more attention to the semantic relationship between words in the sequence itself. Quantitatively, we compare the performance of ALAN and ALAN$_{var}$ with other network models through extensive experiments on English and Chinese datasets. ALAN achieves state-of-the-art results. To avoid performance errors from small test sets, we use the model trained on a small dataset to validate on a large dataset, which also maintains good performance. In addition, the visual images of attention weights show that ALAN can reasonably pay attention to the special information in the input text, which is of great significance in judging the sentiment polarity of sentences. Inspired by the analysis of the error cases, we hope to consider other content and different embedding methods in the Aspect-Location Embedding module to further mine the location memory information. Since neural networks have some instability, our model can be trained to improve its robustness according to the classification theory proposed by Colbrook et al. (2022) Moreover, it is also necessary to optimize the connection between the Aspect-Location Embedding and the Semantic Representation module in future work.

Acknowledgements This work was jointly supported by National Natural Science Foundation of China under Grants (No. 61672022 and No. U1904186), Research Project of Shanghai Graduate Education Association under Grants (ShsgeG202207).

Author Contributions Pengfei Yu: Conceptualization, Methodology, Validation, Writing - original draft, Writing - review & editing. Wenan Tan: Conceptualization, Methodology, Writing - review & editing, Supervision. Weinan Niu: Conceptualization, Methodology. Bing Shi: Writing - review & editing, Supervision.

Data Availability The datasets and the code used in the current study are available from https://github.com/Yflyfly/ALAN. All the datasets gathered from other sources has been publicly available.

Declarations

Consent for Publication Submissions have not been previously published and all co-authors agree to publish.

Competing interests The authors declare that there is no conflict of interest with anybody or any institution regarding the publication of this paper.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., & et al. (2016). Tensorflow: a system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)* (pp. 265–283). https://doi.org/10.48550/arXiv.1605.08695.

Al-Smadi, M., Qawasmeh, O., Al-Ayyoub, M., Jararweh, Y., & Gupta, B. (2017). Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of arabic hotels' reviews. *International Journal of Computational Science and Engineering*. https://doi.org/10.1016/j.jocs.2017.11.006.

Bahdanau, D., Cho, K. H., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *3rd International conference on learning representations*. https://doi.org/10.48550/arXiv.1409.0473.

Berka, P. (2020). Sentiment analysis using rule-based and case-based reasoning. *Journal of Intelligent Information Systems*, *55*(1), 51–66. https://doi.org/10.1007/s10844-019-00591-8.

Bu, J., Ren, L., Zheng, S., Yang, Y., Wang, J., Zhang, F., & Wu, W. (2021). Asap: a chinese review dataset towards aspect category sentiment analysis and rating prediction. In *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies* (pp. 2069–2079). https://doi.org/10.18653/v1/2021.naacl-main.167.

Cambria, E. (2016). Affective computing and sentiment analysis. *IEEE Intelligent Systems*, *31*(2), 102–107. https://doi.org/10.1109/MIS.2016.31.

Chen, Z., Cao, Y., Lu, X., Mei, Q., & Liu, X. (2019). Sentimoji: an emoji-powered learning approach for sentiment analysis in software engineering. In *Proceedings of the 2019 27th ACM joint meeting on european software engineering conference and symposium on the foundations of software engineering* (pp. 841–852). https://doi.org/10.1145/3338906.3338977.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on deep learning*. https://doi.org/10.48550/arXiv.1412.3555.

Colbrook, M. J., Antun, V., & Hansen, A.C. (2022). The difficulty of computing stable and accurate neural networks: On the barriers of deep learning and smale's 18th problem. *Proceedings of the National Academy of Sciences*, *119*(12), 2107151119. https://doi.org/10.1073/pnas.2107151119.

Dai, J., Yan, H., Sun, T., Liu, P., & Qiu, X. (2021). Does syntax matter? A strong baseline for aspect-based sentiment analysis with roberta. In *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies* (pp. 1816–1829). https://doi.org/10.18653/v1/2021.naacl-main.146.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, *10*(7), 1895–1923. https://doi.org/10.1162/089976698300017197.

Fei, H., Li, J., Ren, Y., Zhang, M., & Ji, D. (2022). Making decision like human: joint aspect category sentiment analysis and rating prediction with fine-to-coarse reasoning. In *Proceedings of the ACM web conference 2022* (pp. 3042–3051). https://doi.org/10.1145/3485447.3512024.

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.

Geed, K., Frasincar, F., & Truçsă, M.M. (2022). Explaining a deep neural model with hierarchical attention for aspect-based sentiment classification using diagnostic classifiers. In *International conference on web engineering* (pp. 268–282). https://doi.org/10.1007/978-3-031-09917-5_18.

Hermann, K. M., Kočiskỳ, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., & Blunsom, P. (2015). Teaching machines to read and comprehend. In *Proceedings of the 28th international conference on neural information processing systems* (pp. 1693–1701). https://doi.org/10.48550/arXiv.1506.03340.

Hochreiter, S., Urgen Schmidhuber, J., & Elvezia, C. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.

Karagoz, P., Kama, B., Ozturk, M., Toroslu, I. H., & Canturk, D. (2019). A framework for aspect based sentiment analysis on turkish informal texts. *Journal of Intelligent Information Systems*, *53*(3), 431–451. https://doi.org/10.1007/s10844-019-00565-w.

Kenton, J. D. M.-W. C., & Toutanova, L. K. (2019). Bert: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT* (pp. 4171–4186). https://doi.org/10.48550/arXiv.1810.04805.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 conference on empirical methods in natural language processing* (pp. 1746–1751). https://doi.org/10.3115/v1/D14-1181.

Li, S., Zhao, Z., Hu, R., Li, W., Liu, T., & Du, X. (2018). Analogical reasoning on chinese morphological and semantic relations. In *Proceedings of the 56th annual meeting of the association for computational linguistics (volume 2: short papers)* (pp. 138–143), https://doi.org/10.18653/v1/P18-2023.

Liang, B., Li, X., Gui, L., Fu, Y., He, Y., Yang, M., & Xu, R (2022). Few-shot aspect category sentiment analysis via meta-learning. *ACM Transactions on Information Systems (TOIS)*. https://doi.org/10.1145/3529954.

Liang, B., Su, H., Yin, R., Gui, L., Yang, M., Zhao, Q., Yu, X., & Xu, R. (2021). Beta distribution guided aspect-aware graph for aspect category sentiment analysis with affective knowledge. In *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp. 208–218). https://doi.org/10.18653/v1/2021.emnlp-main.19.

Liu, J., Teng, Z., Cui, L., Liu, H., & Zhang, Y. (2021). Solving aspect category sentiment analysis as a text generation task. In *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp. 4406–4416). https://doi.org/10.18653/v1/2021.emnlp-main.361.

Ma, D., Li, S., Zhang, X., & Wang, H. (2017). Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the 26th international joint conference on artificial intelligence* (pp. 4068–4074). https://doi.org/10.24963/ijcai.2017/568.

Manandhar, S. (2014). Semeval-2014 task 4: aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)* (pp. 27–35). https://doi.org/10.3115/v1/S14-2004.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proc. Int. Conf.Learn. Representations* (pp. 1–12). https://doi.org/10.48550/arXiv.1301.3781.

Ozyurt, B., & Akcayol, M. A. (2021). A new topic modeling based approach for aspect extraction in aspect based sentiment analysis: Ss-lda. *Expert Systems with Applications*, *168*, 114231. https://doi.org/10.1016/j.eswa.2020.114231.

Pennington, J., Socher, R., & Manning, C.D. (2014). Glove: global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1532–1543). https://doi.org/10.3115/v1/D14-1162.

Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., Al-Smadi, M., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O., & et al. (2016). Semeval-2016 task 5: aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)* (pp. 19–30). https://doi.org/10.18653/v1/S16-1002.

Pontiki, M., Galanis, D., Papageorgiou, H., Manandhar, S., & Androutsopoulos, I. (2015). Semeval-2015 task 12: aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)* (pp. 486–495). https://doi.org/10.18653/v1/S15-2082.

Qiu, Y., Li, H., Li, S., Jiang, Y., Hu, R., & Yang, L. (2018). Revisiting correlations between intrinsic and extrinsic evaluations of word embeddings. In *Chinese computational linguistics and natural language processing based on naturally annotated big data* (pp. 209–221), https://doi.org/10.1007/978-3-030-01716-3_18.

Ramaswamy, S. L., & Chinnappan, J. (2022). Recognet-lstm+cnn: a hybrid network with attention mechanism for aspect categorization and sentiment classification. *Journal of Intelligent Information Systems*, *58*, 379–404. https://doi.org/10.1007/s10844-021-00692-3.

Singh, V., Piryani, R., Uddin, A., & Waila, P. (2013). Sentiment analysis of movie reviews: a new feature-based heuristic for aspect-level sentiment classification. In *2013 International mutli-conference on automation, computing, communication, control and compressed sensing (iMac4s)* (pp. 712–717). https://doi.org/10.1109/iMac4s.2013.6526500.

Singh, L. G., & Singh, S. R. (2021). Empirical study of sentiment analysis tools and techniques on societal topics. *Journal of Intelligent Information Systems*, *56*(2), 379–407. https://doi.org/10.1007/s10844-020-00616-7.

Tang, D., Qin, B., Feng, X., & Liu, T. (2016). Effective lstms for target-dependent sentiment classification. In *Proceedings of COLING 2016, the 26th international conference on computational linguistics: technical papers* (pp. 3298–3307). https://doi.org/10.48550/arXiv.1512.01100.

Tripathy, A., Agrawal, A., & Rath, S.K. (2016). Classification of sentiment reviews using n-gram machine learning approach. *Expert Systems with Applications*, *57*, 117–126. https://doi.org/10.1016/j.eswa.2016.03.028.

Varghese, R., & Jayasree, M. (2013). Aspect based sentiment analysis using support vector machine classifier. In *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (pp. 1581–1586). https://doi.org/10.1109/ICACCI.2013.6637416.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*, 5998–6008. https://doi.org/10.48550/arXiv.1706.03762.

Wang, Y., Huang, M., Zhu, X., & Zhao, L. (2016). Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing* (pp. 606–615). https://doi.org/10.18653/v1/D16-1058.

Xiao, L., Hu, X., Chen, Y., Xue, Y., Chen, B., Gu, D., & Tang, B. (2020). Multi-head self-attention based gated graph convolutional networks for aspect-based sentiment classification. *Multimedia Tools and Applications*, 1–20. https://doi.org/10.1007/s11042-020-10107-0.

Xu, H., Liu, B., Shu, L., & Yu, P.S. (2019). Bert post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of NAACL-HLT* (pp. 2324–2335). https://doi.org/10.18653/v1/N19-1242.

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 (volume 1: long papers)* (pp. 2514–2523). https://doi.org/10.18653/v1/P18-1234.

Zhu, L., Zhu, X., Guo, J., & Dietze, S (2022). Exploring rich structure information for aspect-based sentiment classification. *Journal of Intelligent Information Systems*. https://doi.org/10.1007/s10844-022-00729-1.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.