Knowledge-Based Systems 270 (2023) 110559

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

A novel adaptive marker segmentation graph convolutional network for aspect-level sentiment analysis

Pengcheng Wang [a], Linping Tao [a], Mingwei Tang [a,*], Mingfeng Zhao [b], Liuxuan Wang [a], Yangsheng Xu [a], Jiaxin Tian [a], Kezhu Meng [a]

[a] School of Computer and Software Engineering, Xihua University, Chengdu 610039, China
[b] China Mobile Group Design Institute Co., Ltd. Sichuan Branch, Chengdu 610045, China

ARTICLE INFO

Article history:
Received 29 December 2022
Received in revised form 5 April 2023
Accepted 9 April 2023
Available online 14 April 2023

Keywords:
Aspect-level sentiment analysis
Attention mechanisms
Graph convolutional networks
Bi-syntax-aware
Aspect-related graph

ABSTRACT

Aspect-level sentiment analysis is a fine-grained sentiment classification task that aims to identify the sentiment polarity of specific aspects in online reviews. Attention mechanisms and graph convolutional networks have recently been widely used to model associations between aspects and opinion words. However, these methods face challenges in accurately modeling the alignment of aspects and exploiting multiaspect sentiment dependences due to the limitations of dependency trees and the complexities of online reviews. In this paper, we propose a novel adaptive marker segmentation graph convolutional network (AMS-GCN) for aspect-level sentiment analysis. Specifically, the proposed AMS-GCN model enhances the information capacity of words by merging marker information from two datasets and uses an adaptive marker segmentation module to divide different marker information into separate modules. Furthermore, the model employs bi-syntax-aware and semantic auxiliary modules to obtain syntactic and semantic information. The bi-syntax-aware module combines component and dependency trees to capture comprehensive syntactic information. In contrast, the semantic auxiliary module uses an attention score matrix to capture the semantic association information of each word. Moreover, the aspect-related graph is devised to aggregate information about the sentiment of different aspects. Experiments on several benchmark datasets demonstrate that the proposed model achieves state-of-the-art results.

1. Introduction

Due to the growing popularity of social media, sentiment analysis has emerged as a crucial topic in natural language processing [1,2]. Unlike traditional sentiment analysis tasks (i.e., sentence-level and document-level), aspect-level sentiment analysis is more fine-grained. Aspect-based sentiment analysis aims to determine the polarity (e.g., positive, neutral, or negative) of the target aspect in a sentence [3–6]. For example, as shown in Fig. 1, in the sentence "The phone is wonderful, but the battery and the performance are poor", three aspects are discussed: "phone", "battery" and "performance" and their sentiment polarities are positive, negative, and negative respectively. There are three main ABSA tasks, aspect extraction [7–9], aspect detection [10,11], and sentiment classification [12–15]. This paper focuses on sentiment classification for specific aspects.

To effectively tackle the ABSA task, it is essential to model the relationship between aspects and contexts [16,17]. Attention mechanisms have been widely adopted in ABSA models [18–20] and have successfully solved single-aspect terms. However, attention mechanisms are less effective when dealing with phrasal aspect terms and multiple-aspect sentences, as noise can more readily impact performance. For instance, the following is an example: "The falafel was rather overcooked and dried, but the chicken was fine". In this example, the viewpoint word "fine" receives more attention about "falafel", despite referring to a different aspect, namely "chicken". Therefore, it can be challenging for attention mechanisms to solve multifaceted sentences and phrasal aspect terms effectively.

According to some research results in recent years, the relationship between aspect and context can be analyzed based on syntactic structure [21,22], such as graph attention networks (GATs) and graph convolutional networks (GCNs) [23–26]. Syntactic dependency trees can be employed to model long-distance dependencies between words and contexts. However, extracting sentiment dependencies across clauses from syntactic dependency trees can be challenging due to their inherent structure. As illustrated in Fig. 1, the connection between "wonderful" and "poor" inhibits the ability to capture the sentiment polarity of each aspect. Furthermore, although syntactic dependency trees

* Corresponding author.
E-mail address: tang4415@126.com (M. Tang).

https://doi.org/10.1016/j.knosys.2023.110559
0950-7051/

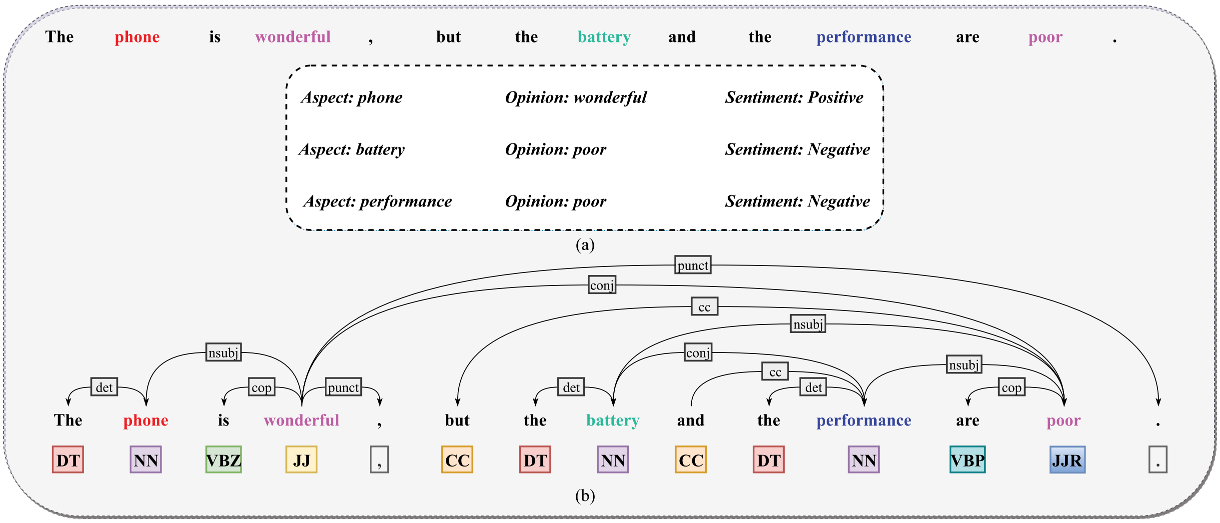


Fig. 1. (a): An example sentence for the ABSA task in a restaurant review, which includes three aspects. (b): Results of dependency tree parsing.

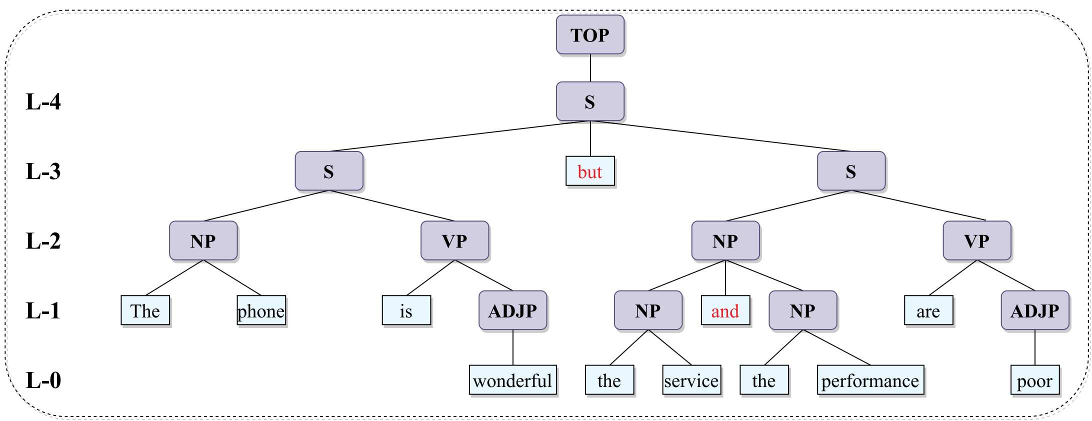


Fig. 2. Component tree for "The phone is wonderful, but the battery and the performance are poor". Rounded rectangles represent phrase types.

create relationships between words, they cannot solve complex relationships such as conditions, oppositions, and joins. Thus, it becomes difficult to model the dependencies between multiple aspects.

In this paper, we propose a novel adaptive marker segmentation of graph convolutional networks (AMS-GCN) to address the abovementioned challenges. First, we combine marked information on words from two identical datasets uniquely. The data merging module (DM) then expands the word-marked information, such as "tok", "aspects", "dephead", "head", "deprel", and "pos", to provide clearer syntactic and semantic information to subsequent processing. Second, we identified that too much word-marked information could lead to information redundancy, negatively impacting the model's performance. To overcome this, we designed an adaptive marker segmentation module (AMS) to segment different word-marked information for the subsequent bi-syntax-aware modules and semantic auxiliary modules. The AMS module dramatically improves the model's applicability by adapting the marker information.

Furthermore, a bi-syntax-aware module is designed to combine syntactic information from dependency trees and component trees [27]. In general, component trees provide a more precise segmentation of phrases and hierarchical structures that facilitate the identification of various aspects and viewpoint words. Through phrase segmentation, it is possible to segment sentences into subclauses. In a hierarchical structure, the relationship between different aspects can be distinguished, and the sentiment dependence of different aspects can be judged. As shown in Fig. 2, the phrase splitter "but" splits the two clauses, "the phone is wonderful" and "the battery and the performance are poor". In the first layer, the word "and" reflects the connected relationship between "battery" and "performance". In the third layer, the term "but" reflects the adversarial relationship between the phone and the other two aspects. In parallel, aspect-related graphs were constructed for internal contexts to aggregate the sentiment information of each aspect. Extracting phrase-level syntactic information from the component tree and its dependency tree's syntactic information, is particularly significant. We also introduced a semantic auxiliary module to extract semantic association information between words. Finally, we designed a fusion module to combine the bi-syntax-aware and semantic auxiliary modules' output.

Our main contributions are as follows:

- We propose a novel adaptive marker segmentation graph convolutional network (AMS-GCN) to address multiple aspects and aspect-specific sentiment dependencies.
- We design the DM module to integrate the word marker information and then adaptively divide the marker information by the AMS module.
- AMS-GCN captures syntactic and semantic information in a dual-channel manner. Moreover, aspect-related sentiment information is aggregated with aspect-related graphs.
- We conduct experiments on four benchmark datasets, and the experimental results demonstrate the validity of our model in ABSA.

The rest of this paper is organized as follows. In Section 2, we introduce related work. Next, Section 3 presents a detailed description of the model characteristics. Then, we compare our model with other models in Section 4. Finally, conclusions and directions for future work are presented in Section 5.

2. Related work

In this section, we present related work on aspect-level sentiment analysis from recent years. First, we introduce some classic aspect-based sentiment analysis methods. Second, we introduce two popular approaches, attentional mechanisms and graph neural networks.

2.1. Aspect-based sentiment analysis

As a more fine-grained entity-oriented sentiment classification task, aspect-based sentiment analysis (ABSA) aims to identify the sentiment polarity of specific aspects [6,28–30]. There are numerous applications of sentiment analysis in the field of natural language processing (NLP) [31,32], such as recommendation systems [33,34] and chatbots [35–38]. Early approaches to sentiment analysis relied on handcrafted features, which made it difficult to model the relationships between aspects and contexts [12,39–41]. To enhance sentiment dictionaries, lexicon-based functions have been developed to perform sentiment analysis [42–44]. Most of these studies used SVMs to construct sentiment classifiers, including bag-of-words methods and sentiment dictionaries [45]. However, it is critical to note that the quality of features has a tremendous influence on the results.

In addition, feature engineering is a labor-intensive process. As a result of outstanding performance in various NLP tasks, neural networks are gaining popularity in sentiment analysis. Classical models, such as recurrent neural networks, recurrent tensor neural networks [46], LSTM [47], and Tree-LSTM [48], are effective in sentiment analysis. However, it is still challenging to distinguish between different emotional orientations at a fine-grained level. Deep learning has been used to construct more nuanced semantic associations between aspects and contexts to address this issue.

2.2. Attention-based approach

Attention-based neural network models' performance is superior to traditional neural networks. Various neural networks have been proposed to implicitly model the semantic relationships between aspects and contexts based on attention. For instance, Wang et al. [18] proposed an attention-based LSTM for identifying aspect-related sentiment information. A multigranular attention network was designed by Fan et al. to capture word interactions between aspects and contexts [49]. Chen et al. [20] used recurrent neural networks to combine long-distance sentiment information with multiple attention mechanisms. For the same reason, Tang et al. [50] employed a deep memory network based on external memory and multi-hop attention. Ma et al. [19] proposed an interactive attention network to learn attention interactively in context and target. Additionally, the pretrained language model BERT [51] has significantly improved NLP tasks, including ABSA. Gao et al. [52] developed three target-dependent variants of the BERT base model, and their effectiveness was demonstrated. Xu et al. [53] explored a novel post-training method to retrain BERT and apply the result for sentiment classification.

2.3. Graph neural network based approach

Recently, the combination of graph neural networks and dependency trees has led to remarkable results in various NLP tasks, such as text classification [54,55] and relationship extraction [56,57]. Graph neural networks are utilized to model dependency trees and encode syntactic information through syntactic dependency trees. Several studies have explored this idea by leveraging syntactic structure information to learn node representations from their neighbors. Zhang et al. [58] used graph convolutional networks on dependency trees of sentences to exploit syntactic information and word dependencies. Wang et al. [25] built ordinary dependency trees into aspect-oriented dependency trees and then encoded this node information using graph attention networks. Tang et al. [59] proposed a dependency graph-enhanced dual-transformer network to enhance the dependency graph representation and the dual-transformer planar representation. Li et al. [26] designed a dual graph convolutional network to jointly consider the syntactic information of the dependency tree and the semantic relevance between words. Tian et al. [60] proposed a type-aware graph convolutional network that explicitly exploits ABSA dependency types and adopts attention layers to integrate different levels of learning. However, since multiple aspects can introduce noise, it is crucial to consider how ABSA tasks can learn about the dependencies among the target aspects and how they affect each other.

Graph convolutional networks and attention mechanisms are shown to improve the model's performance in ABSA. We present a novel model in this paper based on these previous works. Syntactic information and semantic associations are extracted from two-channel convolutional networks with adaptive segmentation in the model. The AMS module is used for the adaptive partitioning of word inputs. The bi-syntax-aware and semantic auxiliary modules encode their respective input information to obtain an enhanced aspectual feature representation. Finally, aspect-related graphs are utilized in the bi-syntax-aware module to emphasize the affective dependencies between different aspects.

3. Methodology

In this section, we present the structure of our proposed AMS-GCN model. The overall architecture of AMS-GCN is shown in Fig. 3. It is composed of four parts: data preprocessing, a word embedding layer, a bi-syntax-aware module, and a semantic auxiliary module.

3.1. Problem description

In this section, we will mathematically define the ABSA task. Suppose the sentence-aspect pair is (S, A), where $S = \{w_1, w_2, \ldots, w_n\}$ denotes a sentence and $A = \{a_1, a_2, \ldots, a_m\}$ denotes a predefined set of aspects, usually a subsequence of the sentence

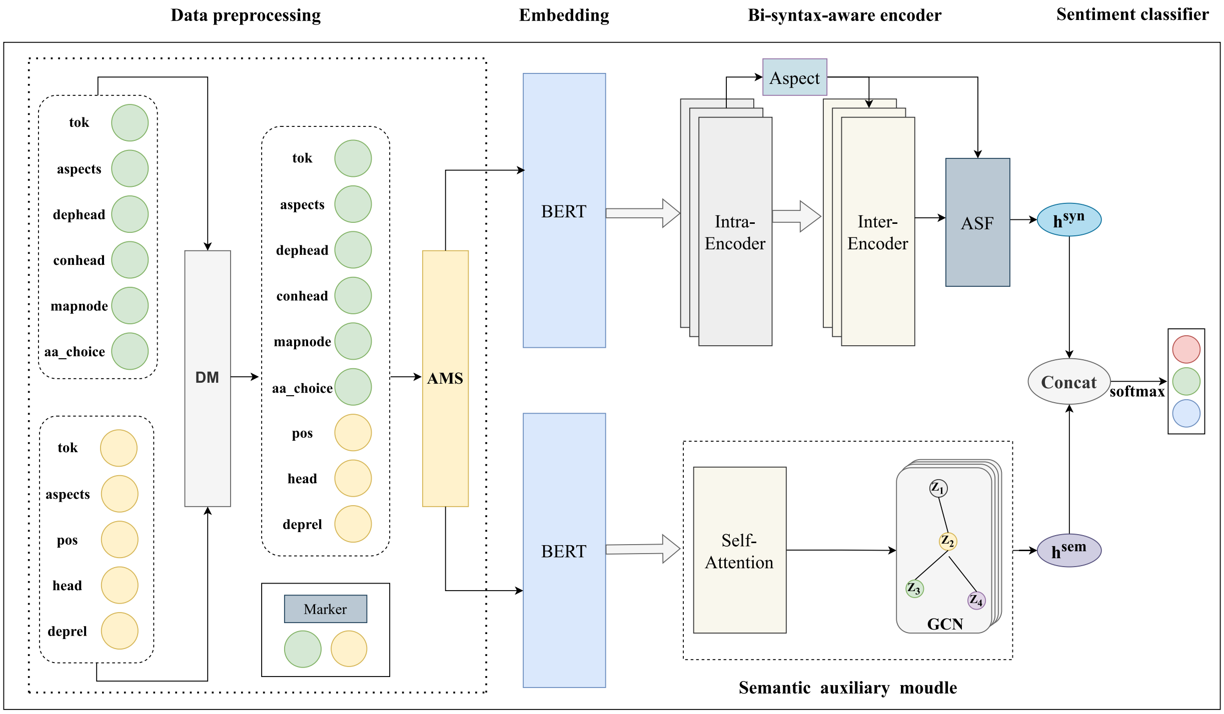


Fig. 3. The overall architecture of AMS-GCN, mainly consists of four components: data preprocessing, word embedding layer, bi-syntax-aware module, and semantic auxiliary module.

S, where n and m are the number of words in S and the number of aspects in A, respectively. It is convenient to consider the multiple-word aspect as a word, denoting the i-th word in S.

The goal of ABSA is to predict the sentiment polarity $y \in \{-1, 0, 1\}$ for each aspect A in sentence S. There are three polarities of sentiment, negative, neutral, and positive, which are denoted by the numbers -1, 0, and 1, respectively.

3.2. Data preprocessing

In this section, we will discuss the basic methods of working with datasets and how to segment datasets adaptively. As a result of the data extension, the dataset has a broader capacity for word information. Two subsequent modules receive the marker adaptive segmentation output: the bi-syntax-aware module and the semantic auxiliary module. Syntactic and semantic information should be explicitly available to the bi-syntax-aware and semantic auxiliary modules.

3.2.1. DM module

As shown in Fig. 3, the DM module merges data labels from two identical datasets, thereby augmenting the data. For instance, some researchers have annotated the Laptops dataset with five categories of information (e.g., "token", "aspects", "pos", "head", "deprel"), while others have used seven categories of information (e.g., "token", "aspects", "postag", "dephead", "pos", "conhead", "maphead"). With the DM module, the Laptops dataset is expanded to have nine categories of marked information (e.g., "token", "aspects", "pos", "head", "deprel", "postag", "dephead", "conhead", "maphead").

Additionally, it can be expressed in mathematical form. We merge the same datasets, D_1 and D_2, that contain different word-marks. It is important to note that we take D_1 as the base during the merging process and keep its format unchanged while adding D_2's data markers.

Simultaneously, the markers in the D_1 dataset are retained when merging the same markers. The merged dataset D's final output is the following input. The process of merging D_1 and D_2 is described in Algorithm 1.

Algorithm 1 Merge information markers from two datasets

Input: Input the original file format of the two datasets D_1 and D_2, and represent the markers in them with M_1 and M_2 respectively.
Output: Output the merged dataset, and store the merge markers.

```
Initialize a list D to store the merged data, and a dictionary dic to save the merged marks.
Open D_1 with data defined as F, open D_2 with data defined as F_1;
for i = 1 → n do
    where n represents the maximum length of D_1;
    if M_{1,i} ∈ D_1 == M_{2,i} ∈ D_2 then
        F_i ← F_i.append(M_{1,i}), where append indicates an add operation;
        dic ← F_i
    else
        F_{1i} ← F_{1i}.append(M_{1,i})
        dic ← F_{1i}
    end if
    D ← append(dic)
end for
return D
```

3.2.2. AMS module

As shown in Fig. 3, the AMS is designed to enable adaptive segmentation of the merged dataset for subsequent modules. First, a threshold range, also known as the adaptive division length, is established based on the function of the submodule. Second, the AMS module can perform adaptive division according to the predetermined threshold range. For instance, consider a Laptops dataset with nine types of label information after the DM module is applied, where we set a threshold range of 5. After applying the AMS module, the dataset is divided into two kinds of tagging information.

Due to the different contents input by each BERT module, it is impossible to feed data D into both BERT modules directly. Instead, the adaptive marker segmentation module (AMS) divides the corresponding markers into upper and lower BERT modules, as shown in Algorithm 2. Then, the input information is adaptively segmented in subsequent modules using the adaptive segmentation algorithm. As a result, the capacity of word information is expanded, and semantic information is enhanced.

Algorithm 2 Adaptive marker segmentation algorithm

Input: Enter the merged marker D and apply adaptive segmentation with the AMS module.
Output: Output the corresponding segmentation information markers T_1 and T_2.
1: Set thresholds h_1 and h_2 to control input markers, initialize the new dictionary $\boldsymbol{dic}_1$ and $\boldsymbol{dic}_2$, and use marks to indicate the keys of the dictionary;
2: d_i denotes the number of input marks;
3: **for** each $i \in D$ **do**
4: **if** $0 \leq d_i \leq$ (control threshold h_1) **then**
5: $T_1 \leftarrow dic_1.append(i)$, where *append* indicates an add operation;
6: **else** $h_1 \leq d_i \leq h_2$ && $i \notin T_1$
7: $T_2 \leftarrow dic_2.append(i)$
8: **end if**
9: **end for**
10: **return** T_1, T_2

3.3. Word embedding layer

Using contextual representations can improve natural language comprehension abilities, thus improving performance. Each word w_i in a sentence is embedded into a continuous low-dimensional vector space using a word embedding matrix. The BERT system has gained popularity in recent years due to its excellent word embedding representations and high performance. The embedding matrix is usually initialized with embeddings from a pretrained model (BERT) to obtain a specific representation of each context and aspect word. The structure of BERT is shown in Fig. 4. Given the target sequence, we first use **BERT − SPC** to construct a BERT-based sequence:

$$BERT\_\,\mathrm{seq} = [\mathrm{CLS}] + \{\boldsymbol{w}_i\} + [\mathrm{SEP}] + \boldsymbol{w}_a + [\mathrm{SEP}] \tag{1}$$

Then, for the input context and aspect words $\boldsymbol{w}_i^s$ and $\boldsymbol{w}_j^a$, we obtain the word embedding vectors $\boldsymbol{o}_i^s \in \mathbb{R}^{d_w}$ and $\boldsymbol{o}_j^a \in \mathbb{R}^{d_w}$ for the context and aspect words, denoted as follows:

$$\begin{aligned} \boldsymbol{o}_i^s &= \mathrm{BERT}\left(\boldsymbol{w}_i^s\right) \\ \boldsymbol{o}_j^a &= \mathrm{BERT}\left(\boldsymbol{w}_j^a\right) \end{aligned} \tag{2}$$

where $\boldsymbol{d_w}$ denotes the dimension of the word embedding.

3.4. Bi-syntax-aware module

After gaining the embedding representation of each word in Section 3.3, we encode this syntactic information with a bi-syntax-aware module. In the bi-syntax-aware module, there are two main parts: the intra-encoder and the inter-encoder. They are applied to create aspect-specific representations by modeling the perceptual context for each aspect. Finally, the corresponding feature representations are fused using the SYF module.

3.4.1. Intra-encoder

This section models the sentiment-aware context of each aspect with a syntactic encoder, and the internal encoder is shown in Fig. 5. Then, aspect-specific representations are created using syntactic information from the parsed dependency tree and the rebuilt component tree. We use this module several times for multiple aspects of the sentence. There are multiple layers of graph attention (MGAT) in our encoder. Each block is composed of multiple graph attention network layers that encode syntactic information in layers with the aid of dependency trees. Additionally, we consider both the component trees' phrase-level syntactic information and the dependency trees' syntactic information. Notably, graph construction makes the MGAT block so powerful.

Graph construction As Fig. 5 illustrates, we follow the syntactic structure of the component tree from top to bottom. The input text is composed of several phrases at each level of the component tree. There are separate semantic blocks within each phrase. For example, in Fig. 1, "The phone is wonderful, but the battery and the performance are poor". We construct relevant graphs based on these phrases. For each layer of the phrase composition, we construct the corresponding adjacency matrix (CTA) to represent the connection between each word. The CTA construction formula is as follows:

$$\mathbf{CTA}_{i,j}^{l} = \begin{cases} 1 & \text{if } w_i, w_j \text{ in same phrase} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

MGAT The multilayer graph attention block encodes syntactical information into word representations in a hierarchical manner. Multiple GAT layers are stacked on top of one another to achieve adaptive matching. The GAT layers aggregate neighbor node information with an attention mechanism. The formula is as follows:

$$\boldsymbol{\alpha}_{ij}^{lz} = \frac{\exp\left(f\left(\boldsymbol{e}_i^{s,l-1}, \boldsymbol{e}_j^{s,l-1}\right)\right)}{\sum_{j' \in \mathcal{N}^l(i)} \exp\left(f\left(\boldsymbol{e}_i^{s,l-1}, \boldsymbol{e}_{j'}^{s,l-1}\right)\right)} \tag{4}$$

$$\boldsymbol{e}_i^{s,l} = \|_{z=1}^{Z}\, \sigma\left(\sum_{j \in \boldsymbol{\mathcal{N}}^l(i)} \boldsymbol{\alpha}_{ij}^{lz} \boldsymbol{W}_V^{lz} \boldsymbol{e}_j^{s,l-1}\right), \tag{5}$$

$$\hat{\boldsymbol{e}}_i^{s,l} = \boldsymbol{FC}\left(\boldsymbol{e}_i^{s,l} + \hat{\boldsymbol{e}}_i^{s,l-1}\right) \tag{6}$$

where $\boldsymbol{\mathcal{N}}^l(i)$ is the set of neighbors of w_i in layers l and $\hat{\mathbf{e}}_{\mathbf{i}}^{\mathbf{s,l}}$ is the final representation of w_i in layer l. $\boldsymbol{W}_V^{lz}$ is the trainable parameter of the zth head of layer l. $\boldsymbol{\alpha}_{ij}^{lz}$ is a normalized attention coefficient computed by the zth attention at layer l. f is a function measuring the correlation of two vectors, $\mathbf{Z}$ is the number of heads of attention and $\boldsymbol{FC}$ is a fully connected feedforward network.

The syntactic dependency tree provides syntactic information about the sentence. To obtain its associated adjacency matrix (DTA), we construct the matrix as follows:

$$\mathbf{DTA}_{i,j}^{k} = \begin{cases} 1 & \text{if } w_i, w_j \text{ link directly in dependency tree} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Furthermore, we have devised methods for combining the dependency and component trees syntax information. For fusion, we construct the adjacency matrix DTA of the syntactic dependency tree and an adjacency matrix CTA of the component tree. There are three different approaches to integration.

A. Dot product operation For each layer in the CTA, only the neighbors in the DTA that are also in the same phrase are considered.

$$\mathbf{FA} = \mathbf{CTA} \cdot \mathbf{DTA} \tag{8}$$

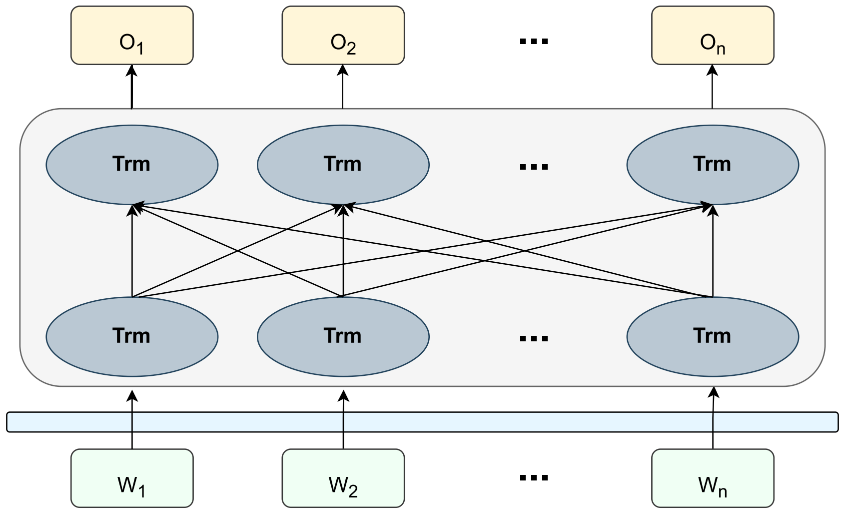


Fig. 4. The overall architecture of BERT.

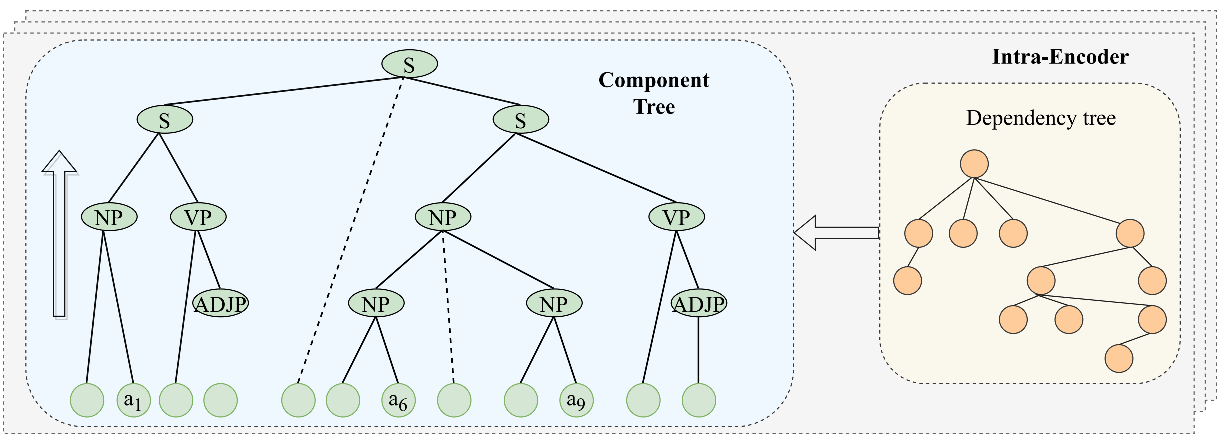


Fig. 5. The overall architecture of the intra-encoder.

B. Additive operations For each layer in the CTA, only consider words from the same phrases and neighbors in the DTA.

$$\mathbf{FA} = \mathbf{CTA} + \mathbf{DTA} \tag{9}$$

C. Conditional addition operations This operation considers the syntactic information of the phrase segmentation items in the CTA and the syntactic information of the clauses in the DTA.

$$\mathbf{FA} = \mathbf{CTA} \oplus \mathbf{DTA} \tag{10}$$

$$\boldsymbol{v}_t = \left[\hat{\boldsymbol{o}}_t^t; \hat{\boldsymbol{e}}_t^t\right] \tag{11}$$

Thus, the intra-encoder contains syntactic information about the dependency and component trees. Where $\hat{\boldsymbol{o}}_t^t$ and $\hat{\boldsymbol{e}}_t^t$ represent the feature representation of the word embedding and the feature representation of the last layer of the intra-encoder, respectively. $\boldsymbol{v}_t$ denotes the final feature representation of the intra-encoder module.

3.4.2. Inter-encoder

In intra-encoder modules, it is impossible to model the links between multiple aspects well because the impact of aspects on each other is ignored. Therefore, we create aspect-related graphs to model cross-aspect connections and aggregate sentiment information between aspects in the inter-encoder module. Moreover, the module's input is based on a precise representation of the intra-encoder. Its structure is shown in Fig. 6. Aspect relations can be revealed by segmentation terms, such as conjunctions. To solve this problem, we developed a regular function RF that returns the phrase splitting term of two aspects. When given two aspects, the first step would be to find their lowest common ancestor in the tree of components. The next step is to look for the internal branch between the two aspects within the subtree. There are only two aspects of information in the tree and very scant irrelevant context. If the internal branch exists, RF returns all the words in the internal branch, otherwise, it returns the words between the two aspects. The formula is as follows:

$$\mathbf{RF}\left(a_i, a_j\right) = \begin{cases} \{w_k\}, & \text{if } \left|\boldsymbol{B_w}\left(a_i, a_j\right)\right| = 0 \\ \boldsymbol{B_w}\left(a_i, a_j\right), & \text{otherwise} \end{cases} \tag{12}$$

where $i < k < j$ and $\boldsymbol{B_w}$ return the words in the internal branches of the two aspects a_i and a_j.

Aspect-dependent graphs As distance increases, the effect between aspects decreases. Considering that all aspects degrade computational performance due to long-distance dependencies, only adjacent aspects are considered. Using the RF function, we obtained the phrase segmentation terms and constructed the aspect-related graph. As shown in Fig. 7, two adjacency matrices are constructed to distinguish the interactions between aspects. In addition to addressing the positive relationship between aspects, it also addresses the negative relationship between aspects. The conclusion of the intra-encoder module's output, $\boldsymbol{v}_t$, is used

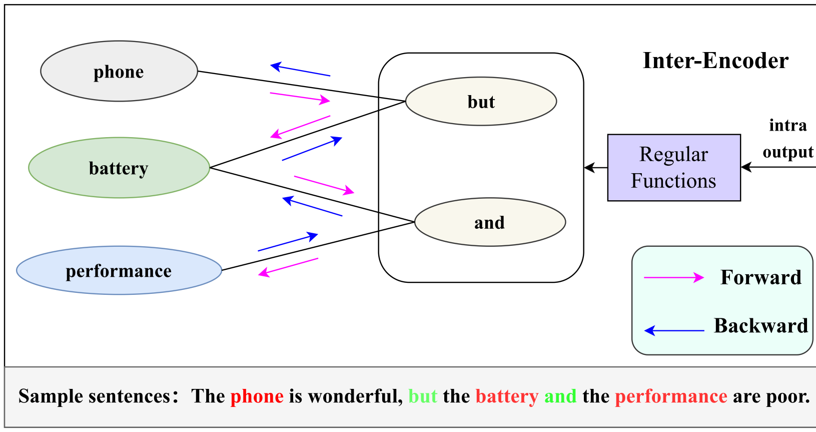


Fig. 6. The overall architecture of the inter-encoder.

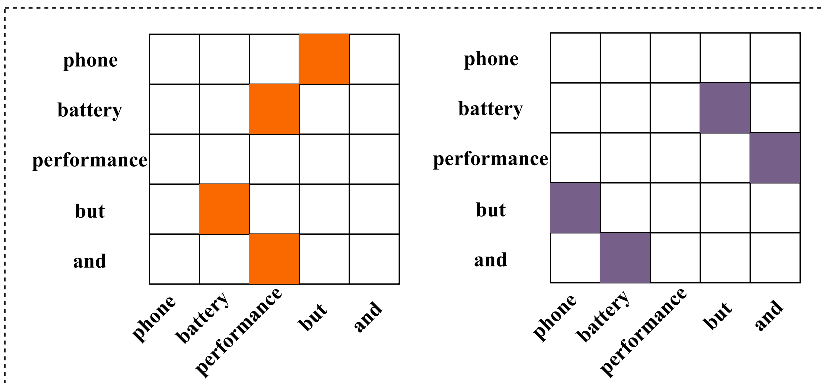


Fig. 7. Aspect-related graphs for distinguishing aspect bidirectional relations.

as input, together with the corresponding phrase segmentation items. MGAT has then been applied again as a relational encoder to obtain an enhanced representation $\boldsymbol{v}_t^a$ of each aspect.

3.4.3. Syntax feature fusion

The final representation of the bi-syntax-aware module is obtained by fusing the output representation of the intra-encoder module with the output representation of the inter-encoder module. The formula is as follows:

$$\boldsymbol{g}_t^{syn} = \textbf{SYF}\left[\boldsymbol{v}_t;\, \boldsymbol{v}_t^a\right] \tag{13}$$

3.5. Semantic encoder

The bi-syntax-aware module may extract syntactic and partial semantic information. Thus, a semantic auxiliary module has been introduced to capture more semantic information. In this module, the attention matrix operates primarily through the self-attention mechanism, and then the semantically relevant terms for each word are extracted from the GCN-encoded attention matrix. A feature of this module is the use of the AMS module, which leads to different input information than the bi-syntax-aware module.

A parallel calculation of the attention score for each element is performed by self-attention. For our SemGCN, we take the attention score matrix $\boldsymbol{M}^{\boldsymbol{sem}}$ as the adjacency matrix of the SemGCN, which can be formulated as:

$$\boldsymbol{M}^{\boldsymbol{sem}} = \mathrm{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{W}_Q \times (\boldsymbol{K}\boldsymbol{W}_K)^{\mathrm{T}}}{\sqrt{d}}\right) \tag{14}$$

where $\boldsymbol{W}_Q$ and $\boldsymbol{W}_K$ are learnable weight matrices, while $\boldsymbol{Q}$ and $\boldsymbol{K}$ represent graphical representations of word embedding layers. In addition, d is the dimension of the input node features. The attention score matrix is obtained using a single self-attention head. After obtaining the adjacency matrix, we encode this adjacency matrix with the GCN module to obtain the final representation of the semantic auxiliary module $\boldsymbol{g}_t^{sem} = \{\boldsymbol{h}_{a_1}^{sem}, \boldsymbol{h}_{a_2}^{sem}, \ldots, \boldsymbol{h}_{a_m}^{sem}\}$. The hidden representation formula is updated as follows:

$$\boldsymbol{h}_i^l = \sigma\left(\sum_{j=1}^{n} \boldsymbol{M}_{ij}\boldsymbol{W}^l\boldsymbol{h}_j^{l-1} + \boldsymbol{b}^l\right) \tag{15}$$

where $\boldsymbol{W}^l$ is a weight matrix, $\boldsymbol{b}^l$ is a bias term, and σ is an activation function.

3.6. Model training

The outputs of the bi-syntax-aware module and the semantic auxiliary module were combined to form the final representation, which was subsequently fed to the softmax activation layer (i.e., the sentiment classifier) to generate probabilities for the three sentiment polarities. Finally, we utilize the cross-entropy loss function to guide the optimization and training of AMS-GCN.

$$\begin{aligned} \boldsymbol{o_t} &= [\boldsymbol{g}_t^{syn}; \boldsymbol{g}_t^{sem}] \\ \boldsymbol{p}(\mathbf{t}) &= \mathrm{softmax}\left(\boldsymbol{W}_p\boldsymbol{o}_t + \boldsymbol{b}_p\right) \end{aligned} \tag{16}$$

$$\mathcal{L}_{\boldsymbol{C}} = -\sum_{(s,a)\in\mathcal{D}}\sum_{c\in\mathcal{C}} \log \boldsymbol{p}(t) \tag{17}$$

where $\boldsymbol{W}_p$ and $\boldsymbol{b}_p$ are parameters of the classifier. $\mathcal{D}$ contains all sentence-aspect pairs, and $\mathcal{C}$ is the collection of distinct sentiment polarities.

Table 1
Statistics of the four datasets.

Dataset	Positive		Neutral		Negative	
	Train	Test	Train	Test	Train	Test
Restaurants	2164	727	637	196	807	196
Laptops	976	337	455	167	851	128
Twitter	1507	172	3016	336	1528	169
MAMS	3380	400	5042	607	2764	329

4. Experiment

4.1. Datasets

We conducted experiments on four benchmark datasets, including the SemEval2014 [61] restaurant reviews (Restaurants) and laptop reviews datasets (Laptops), the ACL14 Twitter [62] datasets, and the MAMS [63] datasets. There are both multi-aspect and single-aspect sentences in the restaurant and laptop datasets. Each sentence in the MAMS dataset contains at least two aspects of sentiment. The statistics of the four datasets are shown in Table 1.

4.2. Experimental settings

In our experiments, Biffine Parser is used for dependency parsing. We represent words with the final hidden state of the pretrained BERT model and fine-tune it in the task. The word embedding dimension is 768, the batch size is 32, and the maximum sentence length is 100. The depth of GCN layers in the model architecture is set to 2, and the dropout rate is set to 0.2. We use the Adam [64] optimizer to optimize all parameters of the training process. The learning rate and BERT learning rate are set to $3e{-}5$ and $2e{-}5$, respectively. To prevent overfitting, we set the random dropout rate to 0.1. In addition, accuracy and Macro-F1 metrics are used to assess the model's performance.

4.3. Baseline methods

In this section, we compare our proposed AMS-GCN model with some other baseline models, including attention-based models, graph neural network (syntactic dependency tree) models, and models based on the BERT method. Details of the baseline models are as follows:

A. Attention-based baseline methods:

- **ATAE-LSTM** [18] is an attention-based long and short-term memory network that explores the links between aspects and context.
- **IAN** [19] employs interactive attention mechanisms to interactively learn the context and aspect-specific representations.
- **RAM** [20] devises an approach that combines multiple attention mechanisms and recurrent neural networks to enhance the representational power of the model.
- **TNet** [65] converts contextual embeddings into target-specific embeddings and extracts significant sentiment features with CNNs.
- **MGAN** [49] proposes a multigrained attention mechanism for capturing interactions between aspects and contexts.
- **BERT** [51] is a pretrained language representation model, that uses a bidirectional transformer for pretraining.

B. Baseline methods based on graph neural networks:

- **TD-GAT** [23] designs a target-dependent graph attention network for sentiment classification leveraging the dependency relationships between words.
- **ASGCN** [58] exploits the syntactic dependency structure of sentences to solve the long-range multiple word dependency problem in sentiment classification.
- **BiGCN** [24] designs an interactive graph convolutional network to learn information from syntactic and lexical graphs.
- **CDT** [21] enhances the contextual embedding representation of bidirectional long-term memory network learning with graph convolutional network GCN.
- **R-GAT** [25] uses a relational graph attention network (GAT) to encode a dependency tree structure with the target as the root.
- **DGEDT** [59] proposes a graph-dependent augmented two-transformer network that interactively learns a planar representation from the transformer and a graphical representation from the dependency graph.

C. BERT-based baseline methods:

- **TD-BERT** [52] proposes a target-related BERT model with multiple variants for sentiment classification.
- **DGEDT+BERT** [59] is equivalent to the DGEDT approach but employs BERT as the aspect-based encoder.
- **R-GAT+BERT** [25] is identical to the R-GAT method but uses BERT as an encoder for word embedding.
- **T-GCN** [60] is a type-aware graphical convolutional network that demonstratively exploits the ABSA dependency types and proposes an attention layer to integrate learning.
- **DM-GCN-BERT** [66] adopts a dynamic multichannel GCN to jointly model syntactic semantic structures for enriched feature representation.
- **DualGCN+BERT** [26] proposes a dual graph convolutional network that considers both the complementary nature of syntactic structure and semantic relevance.

4.4. Comparison results

Experimental results demonstrate that the proposed AMS-GCN outperforms most comparable models, including attention-based, graph network, and BERT models, as shown in Table 2. This model performs better than the previous model proposed for aspect-level analysis. In particular, the proposed AMS-GCN significantly outperforms previous attention-based methods (ATAE-LSTM, IAN, RAM, TNet, MGAN, BERT) when modeling context. This indicates that the two-channel graph neural network can effectively model the context. Compared to the model using BERT individually, AMS-GCN can extract aspect-related information from the embedding representation generated by BERT. Moreover, when exploiting word dependencies and semantic correlations between words, AMS-GCN significantly outperforms GCN-based models (TD-GAT, ASGCN, BiGCN, CDT), illustrating the efficacy of the two-channel graph neural network in combining syntactic and semantic information. Furthermore, AMS-GCN outperforms previous models with heavy GCN (R-GAT, DGEDT) usage, which implies that it can improve GCN models' ability to extract sentiment information. Additionally, AMS-GCN can capture the semantic correlation between words with additional semantic modules, vastly improving aspectual information's representational capability.

In comparing AMS-GCN to other BERT models on four datasets, the AMS-GCN model demonstrated significantly better performance. Moreover, in comparison to DM-GCN-BERT, AMS-GCN was able to achieve comparable performance despite relatively poor results. This is because the AMS-GCN model uses a two-channel GCN to learn emotional information, leading it to achieve the best results in the baseline model.

Table 2
Overall performance of different methods on the four datasets. Acc represents accuracy, and F1 represents the Macro-F1 score. The best results are in bold face, and the second best are underlined. The results with "†" are retrieved from published papers, and "—" indicates not reported. Others are reported based on the open source codes.

Category	Model	Restaurants		Laptops		Twitter		MAMS	
		Acc	F1	Acc	F1	Acc	F1	Acc	F1
Att.	ATAE-LSTM	77.20	–	68.70	–	–	–	–	–
	IAN†	78.60	–	72.10	–	–	–	76.60	–
	RAM	80.23	70.80	74.49	71.35	69.36	67.30	–	–
	TNet†	80.69	71.27	76.54	71.75	74.90	73.60	–	–
	MGAN	81.25	71.94	75.39	72.47	72.54	70.81	–	–
	BERT	85.62	78.28	77.58	72.38	75.28	74.11	–	–
Graph.	TD-GAT†	80.35	76.13	74.13	72.01	72.68	71.15	–	–
	ASGCN†	80.77	72.02	75.55	71.05	72.15	70.40	–	–
	BiGCN	81.97	73.48	74.59	71,84	74.16	73.35	–	–
	CDT†	82.30	74.02	77.19	72.99	74.66	73.66	80.70	79.79
	R-GAT	83.30	76.08	77.42	73.76	75.57	73.82	–	–
	DGEDT†	83.90	75.10	76.80	72.30	74.80	73.40	–	–
BERT.	TD-BERT	85.10	78.35	78.87	74.38	–	–	–	–
	DGEDT+BERT†	86.30	80.00	79.80	75.60	77.90	75.40	–	–
	R-GAT+BERT	86.60	81.35	78.21	74.07	76.15	74.88	–	–
	T-GCN†	87.41	82.23	<u>81.97</u>	<u>78.71</u>	78.03	77.31	<u>83.68</u>	<u>83.07</u>
	DM-GCN-BERT†	<u>87.66</u>	**82.79**	80.22	77.28	<u>78.06</u>	<u>77.36</u>	–	–
	DualGCN+BERT	87.13	81.16	81.80	78.10	77.40	76.02	–	–
Ours	AMS-GCN+BERT	**88.15**	<u>82.32</u>	**82.42**	**79.28**	**78.52**	**77.72**	**85.57**	**85.22**

The improvement of AMS-GCN over baseline is significant at the 0.01 level.

4.5. Ablation study

An ablation study was carried out to further analyze AMS-GCN's impact, and the results are presented in Table 3. It is evident from the table that the model's performance degrades in the absence of adaptive segmentation (AMS). An important reason is that the AMS module refines the input features, leading to more accurate aspect feature representations extracted by the multichannel GCN module. Moreover, the model performs worst on all datasets without the bi-syntax-aware module. The bi-syntax-aware module accurately extracts syntactic feature representations of contexts and aspects, and establishes affective dependencies between aspects and contexts. Therefore, removing this module is not conducive to extracting sentiment information between aspects and contexts. In contrast, the model performs better without the semantic auxiliary module. While semantic auxiliary modules can provide semantic information to the dependencies between contexts and aspects, syntactic information is more critical for extracting the sentiment features of sentences. Additionally, simple semantic information is also derived when extracting sentence syntax information, and additional semantic information can increase the information representation of the model. Furthermore, removing the "intra-encoder" will result in a degradation of the model performance. This shows that the component and dependency trees in the "intra-encoder" are essential for generating convolutional graphs. In particular, using an "intra-encoder" to extract syntactic information about the context and aspects of a sentence can largely facilitate the model in predicting the sentiment polarity of a specific aspect. It should also be noted that removing the "inter-encoder" slightly reduces the model's performance. The aspect-related graphs in the "inter-encoder" can establish connections between aspects, enhancing the feature representation of aspects to improve the performance of multiaspect sentiment prediction.

4.6. Model analysis

This section presents the experimental results of AMS-GCN. First, we examine the impact of adaptive segmentation on the results. Second, we investigate the rationale behind layering the multilayer graphs of attention. Finally, we analyze how the number of aspects in a sentence influences the results.

4.6.1. Impact of the AMS module

The adaptive marker segmentation (AMS) module refines the input form of words, resulting in a more accurate word-dependent representation and the ability to predict aspects of sentiment. To evaluate the impact of the AMS module, we conducted experiments on the Laptops dataset by adjusting the thresholds h. The results are shown in Fig. 8, demonstrating that the accuracy was higher when using the AMS module than without it, regardless of the threshold h. Furthermore, further refining the word input with this module is possible. We also observed that both model accuracies fluctuate when the threshold h is greater or less than the original data length. This indicates that the AMS module divides the marker information too much or too little, which can affect the final feature representation. Additionally, we found that the model performs optimally when the threshold h is equal to the original data length. This suggests that extracting useful sentiment information involves delineating the appropriate marker information.

4.6.2. Impact of MGAT layers

To analyze the effect of MGAT depth, we varied the number of MGAT layers from 1 to 8. We presented the experimental results on the four benchmark datasets in Fig. 9. We observed that two layers of MGAT achieve overall better performance than other numbers of MGAT layers. Therefore, we set the number of MGAT layers to 2 in our experiments. However, the single-layer MGAT layer needed to be improved in accuracy and Macro-F1 scores on all four datasets, indicating its incapability to explore the sentiment features of specific sentences. Moreover, as the number of layers of MGAT increased, the model's performance tended to degrade when the depth of the model exceeded 2. This suggests that excessive MGAT layers can lead to increased parameters, which reduces the learning capability of the model.

4.6.3. Impact of multiple aspects

Based on previous work, the key to the ABSA task is to model the connections between aspects and perspectives to determine the emotional polarity of different aspects. Our experimental results illustrate the effect of multiaspect sentences with various numbers of aspects on the performance of AMS-GCN. To better demonstrate the validity of the proposed model, we compare our

Table 3
Experimental results of the ablation study. The best results are in bold face. “AMS” represents the adaptive segmentation module, “BiSyn” represents the bi-syntax-aware module, “Sem” represents the semantic auxiliary module, “Intra” represents the intra-encoder, and “Inter” represents the inter-encoder.

Model	Restaurants		Laptops		Twitter		MANS	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
AMS-GCN w/o AMS	87.51	81.56	81.75	77.92	77.92	76.97	84.86	84.79
AMS-GCN w/o BiSyn	86.05	80.79	81.02	77.06	77.06	76.65	84.15	84.07
AMS-GCN w/o Sem	87.45	81.76	81.59	77.92	77.76	76.81	84.75	84.67
AMS-GCN w/o Intra	86.92	81.32	81.10	77.70	77.50	76.62	84.34	84.35
AMS-GCN w/o Inter	87.96	82.06	82.12	78.45	78.26	77.21	85.16	85.03
AMS-GCN	**88.15**	**82.32**	**82.42**	**79.28**	**78.52**	**77.72**	**85.57**	**85.22**

The improvement of AMS-GCN over other ablation models is significant at the 0.01 level.

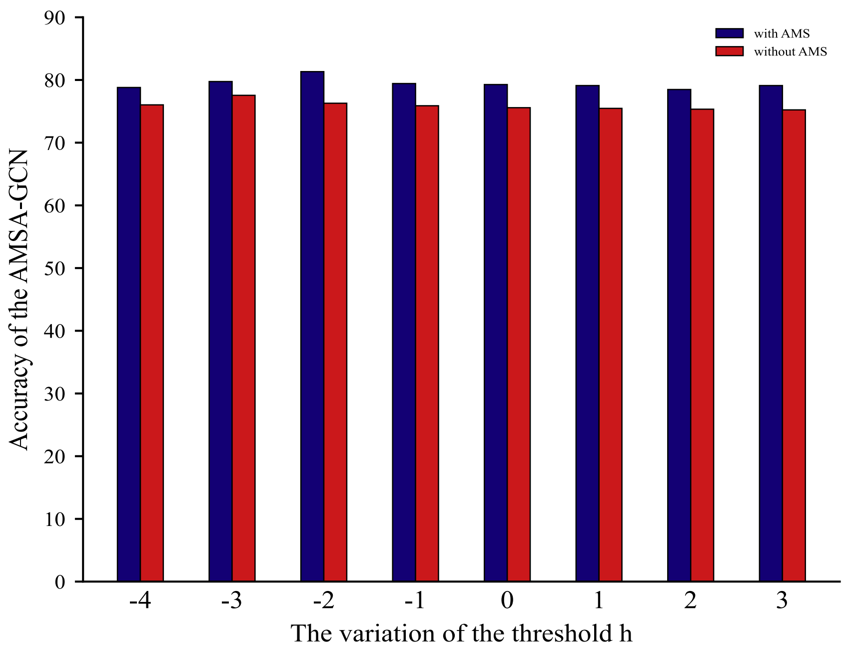
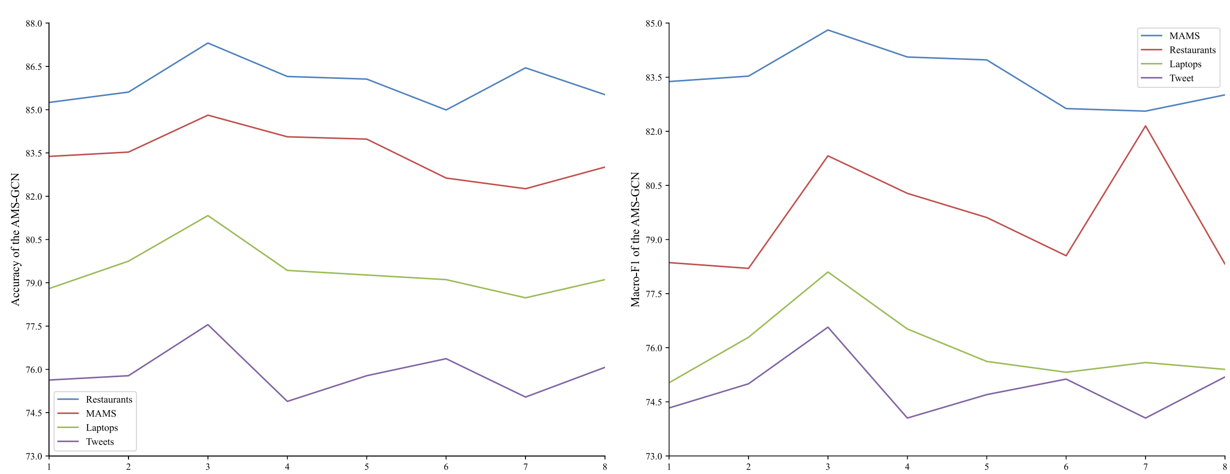


Fig. 8. Accuracy on the Laptops dataset with and without the AMS module.



(a) (b)

Fig. 9. The impact of the number of the MGAT layers of the proposed AMS-GCN. Accuracy and Macro-F1 scores based on different numbers of MGAT layers are reported.

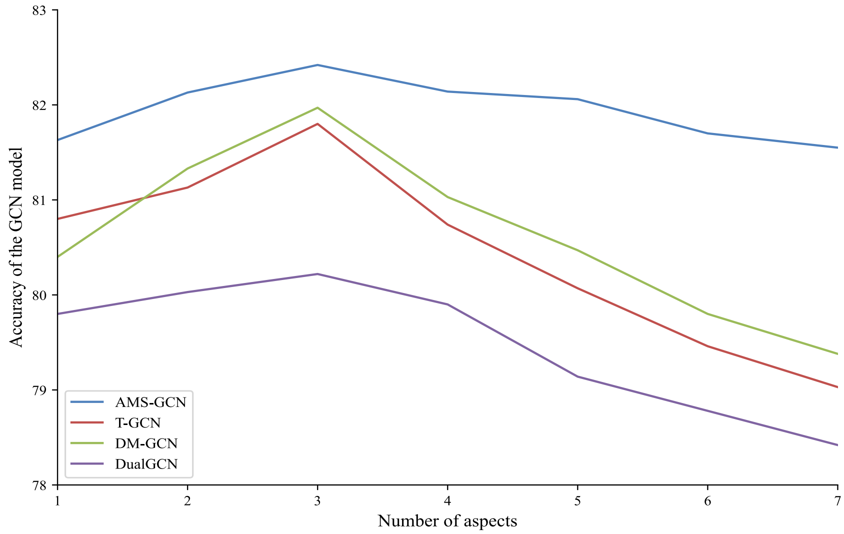


Fig. 10. The impact of multiple aspects in the sentences.

proposed model with other GCN-based models on the Laptops dataset, and the results are shown in Fig. 10. We noticed that when the number of aspects in a sentence was less than 3, the accuracy of all GCN models tended to increase as the number of aspects increased. The possible reason is that when the sentence contains fewer aspects, the constructed dependency graph may be relatively simple, and the GCN-based model is relatively easy to learn. However, for other GCN-based models, the performance significantly decreases as the number of aspects exceeds 3. One apparent reason is that GCN models using dependency trees alone cannot establish sentiment dependencies between aspects when dealing with multiaspect sentences. In contrast, our proposed AMS-GCN model shows slight fluctuations in model accuracy as the number of aspects increases and still achieves good performance. Notably, including component trees and aspect-related graphs enhances the sentiment dependence on a given aspect. We did not analyze sentences with aspect numbers more significant than seven because there are no more examples to make a meaningful comparison. A multiaspect sentence's performance can still be improved over simple sentences. There is the verification of the effectiveness of introducing component trees and aspect-related graphs in AMS-GCN from both sides.

4.7. Attention visualization

We demonstrate how the proposed AMS-GCN model enhances aspectual sentiment prediction by comparing the attention weights of two sentences extracted from restaurant and laptop reviews, as shown in Fig. 11. We selected sentences with more than two aspects and used a darker shade to represent the weight of each word. The diagram indicates that in the first sentence, the contextual word "pleased" is essential in determining the emotional polarity of the three aspects, namely "log on", "WiFi connection", and "battery life". It implies a closer semantic relationship than contextual words, such as "and". Additionally, the composition tree assigns a certain weight to words without emotional polarities, such as "and" and "to". The word "same" is more significant when dividing sentiment polarity into two aspects. Despite the negative word "crappy", the word "same" indicates the sentiment polarity of the aspect. As demonstrated in these multiaspect examples, our proposed AMS-GCN model accurately identifies the sentiment polarity of different aspectual words.

4.8. Case study

As shown in Table 4, we provide several examples of the use of different models to illustrate the advanced capabilities of our proposed AMS-GCN model. In this case, we highlight the words that indicate aspects with yellow and blue colors. In the first example, attention-based approaches such as IAN and MGAN focus on the word "fine". However, complex sentences often fail to connect aspects and opinions through dependency graphs (syntactic dependency trees). In the second sample, the aspect word "music" is relatively distant from the syntax of the viewpoint word, causing the dependency graph-based model to fail. Nevertheless, the RGAT model, based on reconstructing dependency trees, can succeed due to its ability to shorten syntactic distances between viewpoints and aspects. Additionally, the methods based on dependency graphs and attention mostly failed in the third example due to a lack of explicit opinion information. Conversely, the DualGCN model is unaffected because its semantic module (SemGCN) resolves semantic connections between words. Our proposed AMS-GCN can solve multifaceted sentences and accommodate syntactic and semantic information by extracting detailed sentiment features from more complex sentences.

5. Conclusion

In this paper, we propose a novel adaptive marker segmentation graph convolutional network to improve aspect-based sentiment analysis. After integrating the marker information from two identical datasets, we first processed the word marker information and utilized the AMS module to perform adaptive segmentation. Two types of word marker information are fed into the bi-syntax-aware module and the semantic auxiliary module to obtain syntactic and semantic information. In the bi-syntax-aware module, we employ both the composition and syntax trees to get syntactic information. Moreover, we design an aspect-related graph in the inter-encoder to contextually model cross-aspect relationships. Finally, we construct dependency graphs with the attention score matrix in the semantic module to remodel aspects and contexts, thereby obtaining additional semantic association information. The experimental results show that our proposed model achieves state-of-the-art performance on four benchmark datasets. In the future, we plan to include auxiliary information, such as external knowledge, to improve the analytical performance of the ABSA task model.

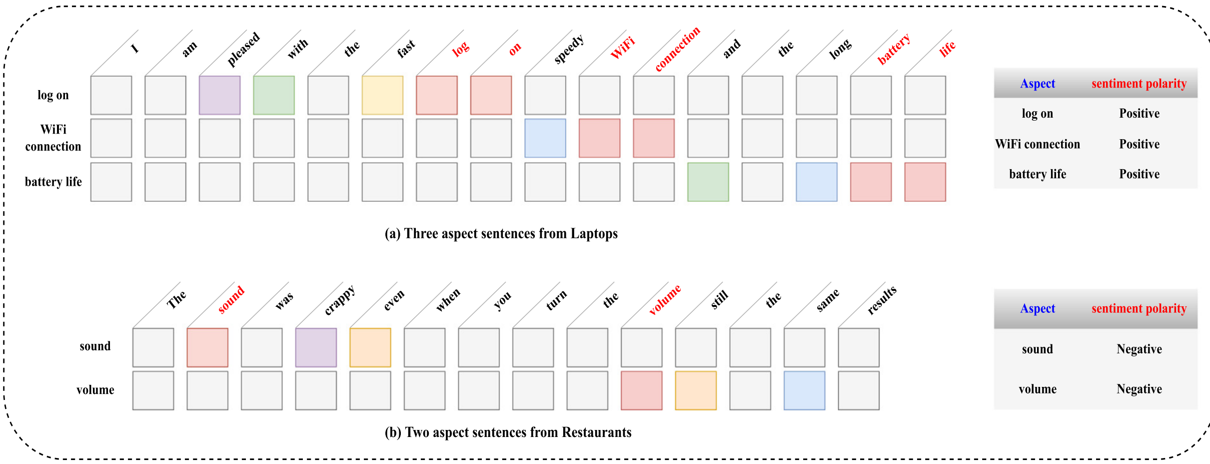


Fig. 11. Attention visualization of two sentences.

Table 4
Case studies of our AMS-GCN model compared with the state-of-the-art baselines. The notations P, N and O represent positive, negative and neutral sentiment, respectively.

#	Review	IAN	MGAN	RGAT	DualGCN	AMS-GCN
1	The falafal was rather over cooked and dried but the chicken was fine!	$(P_{\times}, P_{\checkmark})$	$(P_{\times}, P_{\checkmark})$	$(N_{\checkmark}, P_{\checkmark})$	$(N_{\checkmark}, P_{\checkmark})$	$(N_{\checkmark}, P_{\checkmark})$
2	The music which is sometimes a little too heavy for my taste.	$(N_{\checkmark})$	$(N_{\checkmark})$	$(N_{\checkmark})$	$(P_{\times})$	$(N_{\checkmark})$
3	Entrees include classics like lasagna , fettuccine Alfredo and chicken parmigiana .	$(P_{\times}, P_{\times}, O_{\checkmark}, O_{\checkmark})$	$(P_{\times}, P_{\times}, O_{\checkmark}, O_{\checkmark})$	$(N_{\times}, N_{\times}, P_{\times}, P_{\times})$	$(O_{\checkmark}, P_{\times}, O_{\checkmark}, O_{\checkmark})$	$(O_{\checkmark}, O_{\checkmark}, O_{\checkmark}, O_{\checkmark})$
4	From the speed to the multi touch gestures this operating systems beats Windows easily.	$(O_{\times}, O_{\times}, P_{\checkmark}, N_{\checkmark})$	$(O_{\times}, O_{\times}, P_{\checkmark}, N_{\checkmark})$	$(P_{\times}, O_{\times}, P_{\checkmark}, N_{\checkmark})$	$(P_{\checkmark}, P_{\checkmark}, P_{\checkmark}, O_{\checkmark})$	$(P_{\checkmark}, P_{\checkmark}, P_{\checkmark}, O_{\checkmark})$
5	The food is all-around good, with the rolls usually excellent and the sushi/sashimi not quite on the same level .	$(P_{\checkmark}, O_{\times}, N_{\times})$	$(P_{\checkmark}, O_{\times}, N_{\times})$	$(P_{\checkmark}, P_{\checkmark}, N_{\times},)$	$(P_{\checkmark}, P_{\checkmark}, N_{\times})$	$(P_{\checkmark}, P_{\checkmark}, O_{\checkmark})$

CRediT authorship contribution statement

Pengcheng Wang: Conceptualization, Methodology, Software, Validation, Investigation, Formal analysis, Writing – original draft, Visualization, Writing – review & editing. **Linping Tao:** Validation, Investigation, Software. **Mingwei Tang:** Conceptualization, Investigation, Methodology, Writing – review & editing. **Mingfeng Zhao:** Project administration, Supervision, Visualization. **Liuxuan Wang:** Project administration, Supervision, Formal analysis. **Yangsheng Xu:** Project administration, Supervision, Investigation. **Jiaxin Tian:** Project administration, Supervision. **Kezhu Meng:** Project administration, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work is supported by the Scientific Research Funds project of Science and Technology Department of Sichuan Province, China (Nos. 2019YFG0508, 2019GFW131, 2023G**), the National Natural Science Foundation of China (No. 61902324), Funds Project of Chengdu Science and Technology Bureau, China (Nos. 2023**, 2017-RK00-00026-ZF) and Science and Technology Planning Project of Guizhou Province, China (No. QianKeHeJiChu-ZK[2021] YiBan319).

References

[1] B. Liu, Sentiment analysis and opinion mining, Synth. Lect. Hum. Lang. Technol. 5 (1) (2012) 1–167.
[2] J. Li, E. Hovy, Reflections on sentiment/opinion analysis, in: A Practical Guide To Sentiment Analysis, Springer, 2017, pp. 41–59.
[3] S. Guha, A. Joshi, V. Varma, SIEL: aspect based sentiment analysis in reviews, in: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), 2015, pp. 759–766.
[4] J. Du, L. Gui, Y. He, R. Xu, X. Wang, Convolution-based neural attention with applications to sentiment classification, IEEE Access 7 (2019) 27983–27992.
[5] R. Mao, X. Li, Bridging towers of multi-task learning with a gating mechanism for aspect-based sentiment analysis and sequential metaphor identification, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, (15) 2021, pp. 13534–13542.
[6] R. Mao, X. Li, M. Ge, E. Cambria, MetaPro: A computational metaphor processing model for text pre-processing, Inf. Fusion 86 (2022) 30–43.
[7] G. Chen, Y. Tian, Y. Song, Joint aspect extraction and sentiment analysis with directional graph convolutional networks, in: Proceedings of the 28th International Conference on Computational Linguistics, 2020, pp. 272–279.
[8] S. Tulkens, A. van Cranenburgh, Embarrassingly simple unsupervised aspect extraction, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational

Linguistics, 2020, pp. 3182–3187, http://dx.doi.org/10.18653/v1/2020.acl-main.290, URL https://aclanthology.org/2020.acl-main.290. (Online).

[9] Y. Li, F. Wang, S.-H. Zhong, Training entire-space models for target-oriented opinion words extraction, in: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '22, Association for Computing Machinery, New York, NY, USA, 2022, pp. 1875–1879, http://dx.doi.org/10.1145/3477495.3531768.

[10] R. Oshikawa, J. Qian, W. Wang, A Survey on Natural Language Processing for Fake News Detection, 2018.

[11] N. Vanetik, M. Litvak, S. Shevchuk, L. Reznik, Automated discovery of mathematical definitions in text, in: Proceedings of the 12th Language Resources and Evaluation Conference, 2020, pp. 2086–2094.

[12] L. Jiang, M. Yu, M. Zhou, X. Liu, T. Zhao, Target-dependent twitter sentiment classification, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 2011, pp. 151–160.

[13] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, B. Qin, et al., Learning sentiment-specific word embedding for twitter sentiment classification., in: ACL (1), 2014, pp. 1555–1565.

[14] F. Ma, C. Zhang, B. Zhang, D. Song, Aspect-specific context modeling for aspect-based sentiment analysis, in: CCF International Conference on Natural Language Processing and Chinese Computing, Springer, 2022, pp. 513–526.

[15] B. Wang, L. Ding, Q. Zhong, X. Li, D. Tao, A contrastive cross-channel data augmentation framework for aspect-based sentiment analysis, in: Proceedings of the 29th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Gyeongju, Republic of Korea, 2022, pp. 6691–6704, URL https://aclanthology.org/2022.coling-1.581.

[16] Y. Ma, H. Peng, T. Khan, E. Cambria, A. Hussain, Sentic LSTM: a hybrid network for targeted aspect-based sentiment analysis, Cogn. Comput. 10 (4) (2018) 639–650.

[17] E. Cambria, Q. Liu, S. Decherchi, F. Xing, K. Kwok, SenticNet 7: A commonsense-based neurosymbolic AI framework for explainable sentiment analysis, in: Proceedings of the Thirteenth Language Resources and Evaluation Conference, 2022, pp. 3829–3839.

[18] Y. Wang, M. Huang, X. Zhu, L. Zhao, Attention-based LSTM for aspect-level sentiment classification, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 606–615.

[19] D. Ma, S. Li, X. Zhang, H. Wang, Interactive attention networks for aspect-level sentiment classification, in: Proceedings of the 26th International Joint Conference on Artificial Intelligence, IJCAI '17, AAAI Press, Melbourne, Australia, 2017, pp. 4068–4074.

[20] P. Chen, Z. Sun, L. Bing, W. Yang, Recurrent attention network on memory for aspect sentiment analysis, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 452–461.

[21] K. Sun, R. Zhang, S. Mensah, Y. Mao, X. Liu, Aspect-level sentiment analysis via convolution over dependency tree, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 5679–5688.

[22] A. Murtadha, S. Pan, B. Wen, J. Su, W. Zhang, Y. Liu, BERT-ASC: Auxiliary-Sentence Construction for Implicit Aspect Learning in Sentiment Analysis, 2022.

[23] B. Huang, K. Carley, Syntax-aware aspect level sentiment classification with graph attention networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 5469–5477, http://dx.doi.org/10.18653/v1/D19-1549, URL https://aclanthology.org/D19-1549.

[24] M. Zhang, T. Qian, Convolution over hierarchical syntactic and lexical graphs for aspect level sentiment analysis, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP, 2020, pp. 3540–3549.

[25] K. Wang, W. Shen, Y. Yang, X. Quan, R. Wang, Relational graph attention network for aspect-based sentiment analysis, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2020, pp. 3229–3238, http://dx.doi.org/10.18653/v1/2020.acl-main.295, URL https://aclanthology.org/2020.acl-main.295. (Online).

[26] R. Li, H. Chen, F. Feng, Z. Ma, X. Wang, E. Hovy, Dual graph convolutional networks for aspect-based sentiment analysis, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021, pp. 6319–6329.

[27] S. Liang, W. Wei, X.-L. Mao, F. Wang, Z. He, Bisyn-GAT+: Bi-syntax aware graph attention network for aspect-based sentiment analysis, in: Findings of the Association for Computational Linguistics: ACL 2022, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 1835–1848, http://dx.doi.org/10.18653/v1/2022.findings-acl.144, URL https://aclanthology.org/2022.findings-acl.144.

[28] R. Mao, Q. Liu, K. He, W. Li, E. Cambria, The biases of pre-trained language models: An empirical study on prompt-based sentiment analysis and emotion detection, IEEE Trans. Affect. Comput. (2022).

[29] E. Cambria, R. Mao, S. Han, Q. Liu, Sentic parser: A graph-based approach to concept extraction for sentiment analysis, in: Proceedings of the 2022 International Conference on Data Mining Workshops, Orlando, FL, USA, vol. 30, 2022.

[30] K. He, R. Mao, T. Gong, C. Li, E. Cambria, Meta-based self-training and re-weighting for aspect-based sentiment analysis, IEEE Trans. Affect. Comput. (2022).

[31] L. Zhang, S. Wang, B. Liu, Deep learning for sentiment analysis: A survey, Wiley Interdiscip. Rev.: Data Min. Knowl. Discov. 8 (4) (2018) e1253.

[32] K. Yang, H. Xu, K. Gao, Cm-bert: Cross-modal bert for text-audio sentiment analysis, in: Proceedings of the 28th ACM International Conference on Multimedia, 2020, pp. 521–528.

[33] S. Zhao, W. Wei, D. Zou, X. Mao, Multi-view intent disentangle graph networks for bundle recommendation, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, (4) 2022, pp. 4379–4387, http://dx.doi.org/10.1609/aaai.v36i4.20359.

[34] M. Qiu, X. Huang, C. Chen, F. Ji, C. Qu, W. Wei, J. Huang, Y. Zhang, Reinforced history backtracking for conversational question answering, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, (15) 2021, pp. 13718–13726.

[35] W. Wei, J. Liu, X. Mao, G. Guo, F. Zhu, P. Zhou, Y. Hu, Emotion-aware chat machine: Automatic emotional response generation for human-like emotional interaction, in: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, 2019, pp. 1401–1410.

[36] Q. Li, H. Chen, Z. Ren, P. Ren, Z. Tu, Z. Chen, Empdg: Multiresolution interactive empathetic dialogue generation, 2019, arXiv preprint arXiv:1911.08698.

[37] T. Lan, X.-L. Mao, W. Wei, X. Gao, H. Huang, Pone: A novel automatic evaluation metric for open-domain generative dialogue systems, ACM Trans. Inf. Syst. 39 (1) (2020) 1–37.

[38] W. Wei, J. Liu, X. Mao, G. Guo, F. Zhu, P. Zhou, Y. Hu, S. Feng, Target-guided emotion-aware chat machine, ACM Trans. Inf. Syst. 39 (4) (2021) 1–24.

[39] I. Titov, R. McDonald, Modeling online reviews with multi-grain topic models, in: Proceedings of the 17th International Conference on World Wide Web, 2008, pp. 111–120.

[40] S. Kiritchenko, X. Zhu, C. Cherry, S. Mohammad, NRC-Canada-2014: Detecting aspects and sentiment in customer reviews, in: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), 2014, pp. 437–442.

[41] X. Ding, Y. Zhang, T. Liu, J. Duan, Deep learning for event-driven stock prediction, in: Twenty-Fourth International Joint Conference on Artificial Intelligence, 2015.

[42] D. Rao, D. Ravichandran, Semi-supervised polarity lexicon induction, in: Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009), 2009, pp. 675–682.

[43] V. Perez-Rosas, C. Banea, R. Mihalcea, Learning sentiment lexicons in spanish, in: Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), 2012, pp. 3077–3081.

[44] N. Kaji, M. Kitsuregawa, Building lexicon for sentiment analysis from massive collection of HTML documents, in: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), 2007, pp. 1075–1083.

[45] T. Mullen, N. Collier, Sentiment analysis using support vector machines with diverse information sources, in: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, 2004, pp. 412–418.

[46] R. Socher, A. Perelygin, J. Wu, J. Chuang, C.D. Manning, A.Y. Ng, C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, in: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1631–1642.

[47] A. Graves, Long short-term memory, Superv. Seq. Label. Recurr. Neural Netw. (2012) 37–45.

[48] K.S. Tai, R. Socher, C.D. Manning, Improved semantic representations from tree-structured long short-term memory networks, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Beijing,

China, 2015, pp. 1556–1566, http://dx.doi.org/10.3115/v1/P15-1150, URL https://aclanthology.org/P15-1150.
[49] F. Fan, Y. Feng, D. Zhao, Multi-grained attention network for aspect-level sentiment classification, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 3433–3442.
[50] D. Tang, B. Qin, T. Liu, Aspect level sentiment classification with deep memory network, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Austin, Texas, 2016, pp. 214–224, http://dx.doi.org/10.18653/v1/D16-1021, URL https://aclanthology.org/D16-1021.
[51] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186, http://dx.doi.org/10.18653/v1/N19-1423, URL https://aclanthology.org/N19-1423.
[52] Z. Gao, A. Feng, X. Song, X. Wu, Target-dependent sentiment classification with BERT, Ieee Access 7 (2019) 154290–154299.
[53] H. Xu, B. Liu, L. Shu, P. Yu, BERT post-training for review reading comprehension and aspect-based sentiment analysis, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 2324–2335, http://dx.doi.org/10.18653/v1/N19-1242, URL https://aclanthology.org/N19-1242.
[54] L. Yao, C. Mao, Y. Luo, Graph convolutional networks for text classification, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, (01) 2019, pp. 7370–7377.
[55] L. Huang, D. Ma, S. Li, X. Zhang, H. Wang, Text level graph neural network for text classification, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3444–3450, http://dx.doi.org/10.18653/v1/D19-1345, URL https://aclanthology.org/D19-1345.
[56] Y. Zhang, P. Qi, C.D. Manning, Graph convolution over pruned dependency trees improves relation extraction, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 2205–2215, http://dx.doi.org/10.18653/v1/D18-1244, URL https://aclanthology.org/D18-1244.
[57] H. Peng, L. Xu, L. Bing, F. Huang, W. Lu, L. Si, Knowing what, how and why: A near complete solution for aspect-based sentiment analysis, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, (05) 2020, pp. 8600–8607.
[58] C. Zhang, Q. Li, D. Song, Aspect-based sentiment classification with aspect-specific graph convolutional networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 4568–4578, http://dx.doi.org/10.18653/v1/D19-1464, URL https://aclanthology.org/D19-1464.
[59] H. Tang, D. Ji, C. Li, Q. Zhou, Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 6578–6588.
[60] Y. Tian, G. Chen, Y. Song, Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021, pp. 2910–2922.
[61] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, SemEval-2014 task 4: Aspect based sentiment analysis, in: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEVal 2014), Association for Computational Linguistics, Dublin, Ireland, 2014, pp. 27–35, http://dx.doi.org/10.3115/v1/S14-2004, URL https://aclanthology.org/S14-2004.
[62] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, K. Xu, Adaptive recursive neural network for target-dependent twitter sentiment classification, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2014, pp. 49–54.
[63] Q. Jiang, L. Chen, R. Xu, X. Ao, M. Yang, A challenge dataset and effective models for aspect-based sentiment analysis, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 6280–6285.
[64] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: Y. Bengio, Y. LeCun (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015.
[65] X. Li, L. Bing, W. Lam, B. Shi, Transformation networks for target-oriented sentiment classification, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 946–956, http://dx.doi.org/10.18653/v1/P18-1087, URL https://aclanthology.org/P18-1087.
[66] S. Pang, Y. Xue, Z. Yan, W. Huang, J. Feng, Dynamic and multi-channel graph convolutional networks for aspect-based sentiment analysis, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, 2021, pp. 2627–2636.