Sentiment interaction and multi-graph perception with graph convolutional networks for aspect-based sentiment analysis

Accepted for publication in Knowledge-Based Systems

https://doi.org/10.1016/j.knosys.2022.109840

Sentiment interaction and multi-graph perception with graph convolutional networks for aspect-based sentiment analysis

Qiang Lu[a], Xia Sun[a,*], Richard Sutcliffe[a,b,*], Yaqiong Xing[a], Hao Zhang[c]

[a] *School of Information Science and Technology, Northwest University, Xi'an 710127, China*
[b] *School of Computer Science and Electronic Engineering, University of Essex, Colchester CO43SQ, UK*
[c] *Shaanxi University of Chinese Medicine, Xianyang 712083, China*

Abstract

Graph Convolutional Networks have been successfully applied to aspect-based sentiment analysis, due to their ability to flexibly capture syntactic information and word dependencies. However, most existing graph network-based models only consider the syntactic dependencies between specific aspects and contexts. These cannot capture the internal semantic correlations within aspect-specific phrases and ignore the sentiment interaction relations between different aspects of a sentence. In this paper, we propose a novel graph convolutional network with sentiment interaction and multi-graph perception for aspect-based sentiment analysis. The proposed model considers the complementarity of semantic dependencies and sentiment interactions simultaneously. Specifically, we generate four types of adjacency graph by integrating the internal semantic correlations between aspect phrases and linking the sentiment interaction relations among different aspects. Adjacency graphs are used to construct graph convolution neural networks to enrich aspect-centric dependencies and enhance the capability of context-awareness. In addition, we construct a multi-graph perception mechanism to capture the specific dependency information that cannot be captured between different graphs and hence reduce the overlapping information. Experimental results on five publicly-available datasets demonstrate that our proposed model outperforms state-of-the-art methods and achieves the best performance in terms of accuracy and macro-F1.

*Corresponding author

Email addresses: nwulq@stumail.nwu.edu.cn (Qiang Lu), raindy@nwu.edu.cn (Xia Sun), rsutcl@nwu.edu.cn (Richard Sutcliffe), xkpg@nwu.edu.cn (Yaqiong Xing), 1271009@sntcm.edu.cn (Hao Zhang)

Keywords: Graph Convolutional Network, Internal semantic correlations, Sentiment interaction relations, Multi-graph perception, Aspect-based sentiment analysis

1. Introduction

Sentiment analysis is the process of analyzing, processing, summarizing and reasoning about subjective texts [1, 8, 39, 49]. With the rapid development and wide application of social media, large numbers of subjective texts appear on the Internet every day. Such texts contain information in many fields such as education, economics, and electronic commerce, and analysis of their sentiment is a research hotspot in the field of natural language processing [5, 34, 50, 57]. For example, in the field of e-commerce, sentiment analysis of reviews can help customers choose goods through their evaluation by others. Meanwhile, businesses can also improve goods and services according to user preferences under the guidance of sentiment analysis. In addition, AI systems are becoming more and more accurate, but at the same time, less and less transparent. Therefore, in the field of sentiment analysis, researchers are focusing more on the ability of systems to give the exact reasons for their judgements [9, 10].

In recent years, aspect-based sentiment analysis has attracted extensive attention. This is a fine-grained task that aims to predict the sentiment polarity (e.g., positive, negative, neutral) towards each specific aspect in a given sentence [35, 38, 54]. Aspect-based sentiment analysis is needed because a single sentence may refer to several aspects of a product or service, say, and express different sentiments for each aspect. For such sentences, it is obviously unreasonable to analyze the sentiment of the whole sentence or paragraph rather than each aspect. For example, in the sentence "The spaghetti is delicious, but the waiter is rude.", the sentiment word $delicious$ modifies the aspect $spaghetti$, and $rude$ modifies the aspect $waiter$. For $spaghetti$, customers show positive sentiment, but for $waiters$ they show negative sentiment. The goal of aspect-based sentiment analysis is to capture the aspects [19, 32, 36] and identify the sentiment expressed towards each aspect. In particular, it is divided into two tasks: aspect term sentiment analysis [33, 61] and aspect category sentiment analysis [24, 48]. Aspect term analysis identifies aspect entities in a sentence, and aspect category analysis infers the category of each aspect entity, chosen from a predefined set. For instance, in the sentence "A group of friendly staff, the pizza is not bad, but the beef cubes are not worth the money", the aspect terms are *staff*, *pizza* and *beef cubes*, the aspect categories are *service* and *food*, and the sentiment words are *friendly*, *not bad* and *not worth*. Aspect term *staff* implicitly refers to the aspect category *service*, while aspect terms *pizza* and *beef cubes* implicitly refer to the aspect category *food*. Therefore, we infer that the text expresses a positive sentiment towards the *service* category and a negative sentiment towards the *food* category.

Previous studies have introduced machine learning into aspect-based sentiment analysis due to its ability to capture feature representations [2, 6], but the quality of model classification results depends on the quality of manually annotated data, which is labor-intensive to produce. With the increasing application of neural networks to natural language processing (NLP) tasks [53], recurrent neural networks (RNNs) and the attention mechanism have been introduced into aspect-based sentiment analysis due to their ability to capture both sequential information and contextual semantic information [3]. On the other hand, RNN-based models with attention lack the ability to capture syntactic information and word dependencies. Therefore, graph convolutional networks (GCNs) have been applied to aspect-based sentiment analysis [63]. GCNs show excellent performance in capturing syntactic information and word dependencies [7, 12], but most graph network-based methods only consider the syntactic dependencies between specific aspects and contexts, for example whether there are dependencies between two nodes. If there is a dependency, the node value at the corresponding position of the adjacency graph is set to one, otherwise it is set to zero. However, in our view, the nodes should not only contain syntactic dependencies, but in addition (1) the sentiment relationship between aspect and non-aspect words, (2) the semantic dependencies within aspect phrases, and (3) the sentiment interactions between different aspects should be fully considered.

In this paper, we explore the internal semantic correlations within aspect phrases and also sentiment interaction relations within different aspects, and propose a Sentiment Perception Graph Convolutional Network (SPGCN) incorporating four types of graph, together with a multi-graph perception mechanism for aspect-based sentiment analysis.

On the one hand, aspect phrases are often recognized as an indivisible whole, thus ignoring the internal semantic information among aspect words. Therefore, we capture the semantic information by deriving the correlations among aspect phrases to enhance aspect-centric dependencies. At the same time, in order to further enrich the contextual dependencies, we integrate the contextual information of specific aspects into the syntactic dependency tree.

On the other hand, different aspects of the sentence are often considered to exist independently. In reality, there are abundant associations between different aspects. We generate sentiment interaction graphs by integrating the sentiment relations into the syntactic dependencies to fully utilize such sentiment associations. In addition, because the related entities of each word could be in different regions in a sentence, we expect the model to learn syntactic information and dependencies by interaction between graphs. In order to reduce the overlap of information, we propose a multi-graph perception mechanism to capture the dependency information which cannot otherwise be captured between different graphs. Experiments on five publicly-available datasets show that our proposed model can determine the internal sentiment correlations within aspect phrases as well as the sentiment interactions among different aspects.

The main contributions can be summarized as follows:

- We propose a graph convolutional network which integrates internal semantic correlations and links the sentiment interaction relations to enhance aspect-centric dependencies and the capability of context-awareness.

- A multi-graph perception mechanism is constructed to capture the specific dependency information which cannot be captured between different dependency graphs and thus we reduce the overlap of information.

- Experimental results on five publicly-available datasets demonstrate that our proposed model outperforms state-of-the-art methods and verify the effectiveness of our model.

The remainder of this paper is organized as follows. After introducing previous work in Section 2, we propose the SPGCN architecture comprising a graph convolutional network with sentiment interaction graphs and a multi-graph perception mechanism in Section 3. The experimental details and analysis are then described in Section 4. Finally, we summarize our work and provide an outlook of future work in Section 5.

2. Related work

Machine learning methods were introduced into aspect-based sentiment analysis due to their ability to capture the features based on semantic information and syntactic dependency [6, 48, 55]. Recently, deep learning methods based on neural networks have been widely applied to aspect-based sentiment analysis and great achievements have been reported [14, 15, 41].

Previous methods have introduced recurrent neural networks into aspect-based sentiment analysis, due to their ability to flexibly capture sequential information [3]. Wang et al. [52] integrated RNNs and conditional random fields into a unified framework for explicit aspect term and opinion term co-extraction. However, the RNNs were insufficient to solve the lack of semantic information and gradient disappearance, so the long short-term memory network (LSTM) was applied in aspect-based sentiment analysis [18, 31, 58]. Tang et al. [43] employed two LSTMs to model the left context and right context for given aspects. Ruder et al. [37] introduced a hierarchical bidirectional LSTM that was able to leverage both intra- and inter-sentence relations. On the other hand, sequence models such as LSTM cannot perform parallel computing, and their current information depends on previous information. Researchers attempted to use the attention mechanism for aspect-based sentiment analysis [16, 28, 40, 4]. For example, Wang et al. [56] proposed a novel model which adopted two ways to take into account aspect information using attention. Based on Wang et al. [56], Tay et al. [46] adopted circular convolution of vectors for performing word-aspect fusion. Ma et al. [30] proposed a hierarchical attention model and extended the classic LSTM cell with components accounting for integration with external knowledge. The models based on RNN and the attention mechanism showed great performance in capturing global semantic information, but were weak in local semantic information. Convolutional neural networks (CNNs) were applied to aspect-based sentiment analysis due to their ability to capture local semantic information [24]. Xue et al. [59] constructed novel Gated Tanh-ReLU Units at the top of convolution layers to extract aspect-specific sentiment information. Huang et al. [20] designed a novel parameterized filter and parameterized gate for aspect-based sentiment analysis. These previous methods based on deep learning have achieved many advances, but they lack the ability to capture syntactic information and word dependencies. Therefore, graph convolutional networks were introduced into aspect-based sentiment analysis.

GNNs have achieved great performance in NLP tasks such as text classification [22, 62] and question answering [17, 23]. In aspect-based sentiment analysis, Zhang et al. [63] introduced GCNs for the first time. Zhou et al. [65] developed the syntactic dependency tree and commonsense knowledge strategies to enhance the representation of the sentence concerning a given aspect. To enhance the ability to capture implicit aspect sentiments, Cai et al. [7] used a lower-level GCN for relationship modeling and a higher-level GCN to capture the relationship between aspects. Chen et al. [12] built different contextual features by considering the direction forwards or backwards from a word in context to other words in a sentence. To address the issues of noise and instability in dependency trees, Tan et al. [45] introduced the dependency graph enhanced dual-transformer to fuse the flat representations learnt by the transformer and the graph-based representations learnt based on the dependency graph. The earlier methods only leveraged dependency relations without considering their dependency types, but Tian et al. [47] proposed a type-aware GCN which incorporated word relations and their dependency types to comprehensively learn from dependency parsing results. Overall, previous GCN models take into account syntactic dependencies between a specific aspect and its context, but they are not able to use semantic correlations within aspect-specific phrases, or sentiment interaction relations among different aspects of a sentence. To address these shortcomings, we propose a novel graph convolutional network with sentiment interaction graphs and a multi-graph perception mechanism for aspect-based sentiment analysis.

3. Proposed approach

In this section, the proposed SPGCN model is described in detail (Fig. 1). SPGCN contains three components: sentiment interaction graphs, a multi-graph perception mechanism, and a GCN model. The model first generates the interaction graph by deriving internal semantic information from aspect-specific phrases. Then, contextual information is fused into syntactic dependencies to build the enhancement graph. Finally, the sentiment interaction relations within different aspects are integrated to construct a sentiment interaction graph. For the multi-graph perception mechanism, the sentiment interaction graphs are fused to capture the specific dependency information that cannot be captured between different graphs, and the perception coefficient is set to reduce the overlapping information. The hidden representation, sentiment interaction graphs and multi-graph perception are input into the GCN model. The hidden representation of each node is updated through a graph convolution operation with a normalization factor.

3.1. Notation definition

Some notations are introduced to facilitate the subsequent descriptions: $S = \{s_1, s_2, \cdots, a_{k+1}, \cdots, a_{k+m}, \cdots, s_n\}$ denotes an input sentence which contains a corresponding aspect $X = \{a_{k+1}, \cdots, a_{k+m}\}$. In particular, aspect phrases contain multiple aspect words. In $A = \{a_{11}, a_{12}, \cdots, a_{1t}, \cdots, a_{21}, a_{22}, \cdots, a_{st}\}$, an aspect contains different aspect words, where a_{ij} denotes the j-th aspect word of the i-th aspect. We embed each word token into a low-dimensional real-valued vector space with an embedding matrix $E = \{e_1, e_2, \cdots, e_n\}$, where $e_n \in \mathbb{R}^{d_{emp} \times |N|}$. d_{emp} denotes the dimension of the word embedding, and N indicates the size of the vocabulary. Given the word embedding of the sentence, a bidirectional LSTM is constructed to produce hidden state vectors: $H^c = \{h_1^c, h_2^c, \cdots, h_k^c, \cdots, h_{k+m}^c, \cdots, h_n^c\}$.

$$\begin{aligned} \overrightarrow{h_t^c} &= LSTM(\overrightarrow{h_{t-1}^c}, \overrightarrow{e_t}) \\ \overleftarrow{h_t^c} &= LSTM(\overleftarrow{h_{t-1}^c}, \overleftarrow{e_t}) \end{aligned} \tag{1}$$

$$h_t^c = [\overrightarrow{h_t^c}; \overleftarrow{h_t^c}] \tag{2}$$

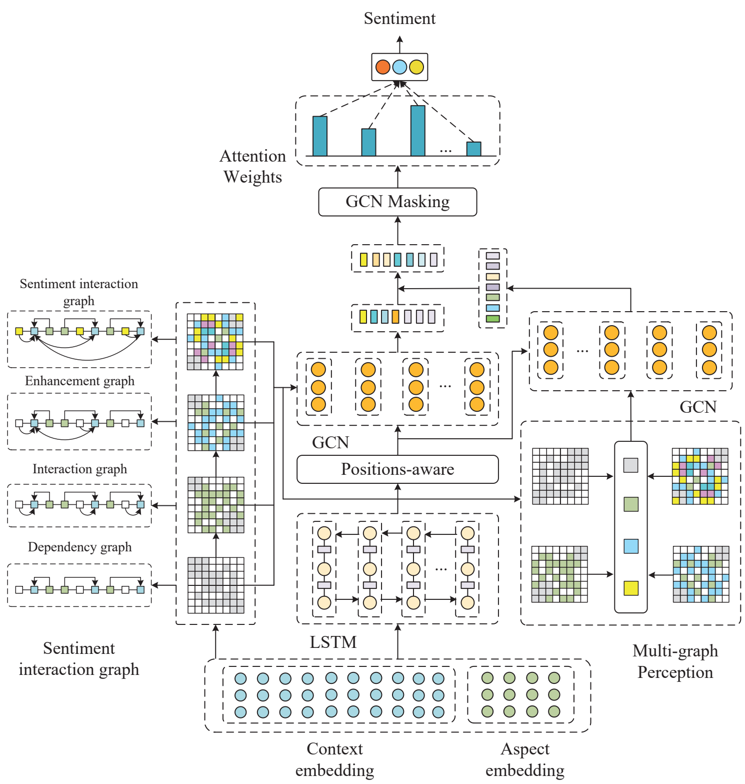

Fig. 1. The architecture of the proposed Sentiment Perception Graph Convolutional Network (SPGCN). Firstly, the sentiment interaction graph, which contains four types of adjacency graph, is generated by integrating the internal semantic correlations between aspect phrases and linking the sentiment interaction relations among different aspects. Then, the multi-graph perception mechanism based on sentiment interaction graphs is constructed to learn syntactic information and dependencies by interaction between the various graphs as a pipeline. Finally, the GCN is used to predict sentiment polarity.

where $h_t^c \in \mathbb{R}^{2d_h}$ is a hidden state vector at time step t from the bidirectional LSTM, d_h is the dimensionality of a hidden state vector output by a unidirectional LSTM, and e_t represents a word embedding.

3.2. Sentiment interaction graph

As demonstrated in Fig. 1, the embedding matrix $E = \{e_1, e_2, \cdots, e_n\}$ is input into three components: sentiment interaction graphs, a multi-graph perception mechanism, and an LSTM. We first capture the internal semantic correlations and sentiment interaction information through the sentiment interaction graph. Specifically, we first generate an interaction graph by calculating relative position information among the aspect-specific phrases. Then, contextual information concerning a specific aspect is fused into the interaction graph to build an enhancement graph. Finally, the sentiment interaction information among different aspects is consolidated to construct the sentiment interaction graph.

The architecture of the sentiment interaction graphs is shown in Fig. 2. In the sentence “Macbook notebooks quickly die out because of their short battery life, as well as the many unknowable background programs”, there are three aspects, “Macbook notebooks”, “battery life” and “background programs”. Firstly, these aspect phrases are composed of different aspect words. In particular, “notebooks”, “life” and “programs” are the headwords of three aspects, and “Macbook”, “battery” and “background” modify the headwords. This implies that there are internal semantic correlations between aspect words in specific aspect phrases. Then, three sentiment words “die out”, “short” and “unknowable” modify aspects “Macbook notebooks”, “battery life” and “background programs” respectively. Finally, concerning the complete sentence “Macbook notebooks quickly die out because of their short battery life, as well as the many unknowable background programs”, aspects “battery life” and “background programs” have sentiment effects on “Macbook notebooks”, and there are sentiment interaction relations between the three aspects.

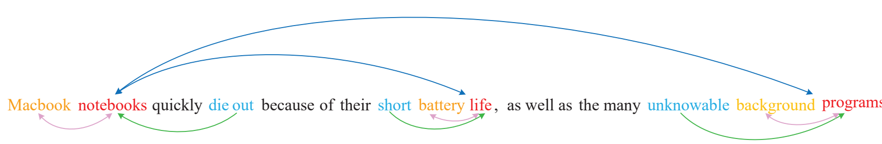

Fig. 2. The architecture of a sentiment interaction graph. The purple bidirectional link represents the internal semantic correlations within aspect phrases, the blue bidirectional link represents the sentiment interaction between different aspects, and the green unidirectional link represents the influence of sentiment words towards aspects.

In contrast to the graphs of previous models, we construct a sentiment interaction graph which not only considers the internal semantic correlations among adjacent nodes based on syntactic dependencies, but also fuses the contextual information towards specific aspects in the graph. Meanwhile, the sentiment interaction relations of different aspects are also integrated into the graph.

3.2.1. Dependency graph

Inspired by previous GCN-based models [27], we construct the original dependency graph based on the dependency tree[1]:

$$D_{i,j}^{G} = \begin{cases} 1, & \text{if } i = j \text{ or } s_i \text{ and } s_j \text{ in the dependency tree} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $D_{i,j}^{G} \in \mathbb{R}^{n \times n}$ denotes the graph matrix, and s_i represents the i-th contextual word in the sentence.

3.2.2. Interaction graph

The interaction graph is produced based on $D_{i,j}^{G}$, and it aims to capture the internal semantic relations of different aspect words in aspect phrases. In particular, we generate the interaction graph by deriving relative position information among the aspect-specific phrases:

$$D_{i,j}^{I} = \begin{cases} 1 + 1/\left(p_a + 1\right), & \text{if } s_i \text{ and } s_j \in (A \cap T) \\ 1 + 1/\left(\left|p_a - j\right| + 1\right), & \text{if } s_i \in (A \cap T) \\ 1 + 1/\left(\left|p_a - i\right| + 1\right), & \text{if } s_j \in (A \cap T) \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $D_{i,j}^{I}$ represents the interaction graph towards aspect phrases, $|.|$ denotes absolute value, s_i and s_j are the contextual words, A denotes the aspect phrases, T denotes the dependency tree, and p_a is the start position of the specific aspect.

3.2.3. Enhancement graph

To enhance the contextual-awareness of aspect-specific phrases, the contextual dependency information towards aspect phrases is fused into an interaction graph to

[1]We use the spaCy toolkit to construct the dependency tree: `https://spacy.io/`

generate an enhancement graph:

$$D_{i,j}^{E}=\begin{cases} D_{i,j}^{G}+1/\left(p_a+1\right)*D_{i,j}^{I}, & \text{if } D_{i,j}^{G} = 1\\ 1/\left(\left|p_a-j\right|+1\right), & \text{if } s_i\in A\\ 1/\left(\left|p_a-i\right|+1\right), & \text{if } s_j\in A\\ D_{i,j}^{I}, & \text{otherwise}\end{cases} \tag{5}$$

where $D_{i,j}^{E}$ represents the enhancement graph which focuses the contextual information towards aspect phrases, and $D_{i,j}^{G}$ and $D_{i,j}^{I}$ represent the dependency graph and the interaction graph. Through the enhancement graph, we not only capture the semantic correlations between different words in aspect phrases, but also obtain contextual dependency information for specific aspects.

3.2.4. Sentiment interaction graph

As shown in Fig. 2, “Macbook notebooks quickly die out because of their short battery life, as well as the many unknowable background programs”, there are sentiment interaction relations between “Macbook notebooks”, “battery life” and “background programs”. To capture sentiment interactions of the different aspects, we construct the sentiment interaction graph:

$$A_{i,j}^{C}=1+1/(\left|p_a-p_o\right|+1)*D_{i,j}^{C} \tag{6}$$

$$D_{i,j}^{C}=\begin{cases} 1/\left(p_o\sum_{n=1}^{N}\frac{p_o}{p_o+1}+1\right), & \text{if } s_i \text{ and } s_j\in A\\ 1/\left(p_o\sum_{n=1}^{N}\frac{p_o}{p_o-j}+1\right), & \text{if } s_i\in A \text{ and } s_j\in O\\ 1/\left(p_o\sum_{n=1}^{N}\frac{p_o}{p_o-i}+1\right), & \text{if } s_j\in A \text{ and } s_i\in O\\ 0, & \text{otherwise}\end{cases} \tag{7}$$

where $A_{i,j}^{C}$ represents the sentiment interaction graph which directs attention towards the different aspects, O is the set of phrases corresponding to other aspects, and p_o is the start position of the other aspects in a sentence. To capture various semantic and syntactic dependencies, we further improve and perfect the sentiment interaction graph, i.e., $A_{i,j}^{C}$ = $A_{j,i}^{C}$. The process of constructing the sentiment interaction graphs is shown in Algorithm 1.

Algorithm 1 The pseudocode for constructing sentiment interaction graphs

Input: a input sentence $S = \{s_1, s_2, \cdots, s_{n-1}, s_n\}$, which contains the aspects $X = \{a_k, \cdots, a_{k+m}\}$; The specific aspect $A^s = \{a_1^s, \cdots, a_{n-1}^s, a_n^s\}$ and the other aspects $A^o = \{a_1^o, a_2^o, \cdots, a_n^o\}$; The dependency tree of the input sentence T

Output: The sentiment interaction graphs A^C.

while (T <maximum number of iterations) **do**
  for $i = 1 \to n; j = 1 \to n$ **do**
    ▷ The process of constructing the dependency graph
    if i = j or s_i and s_j in T **then**
      $D_{i,j}^G \leftarrow 1$
    else
      $D_{i,j}^G \leftarrow 0$
    ▷ The process of constructing the interaction graph
    if s_i and s_j in T and A^s **then**
      $D_{i,j}^I \leftarrow 1 + 1/(p_a + 1)$
    else if s_i in T and A^s **then**
      $D_{i,j}^I \leftarrow 1 + 1/(|p_a - j| + 1)$
    else if s_j in T and A^s **then**
      $D_{i,j}^I \leftarrow 1 + 1/(|p_a - i| + 1)$
    else
      $D_{i,j}^I \leftarrow 0$
    ▷ The process of constructing the enhancement graph
    if $D_{i,j}^G = 1$ **then**
      $D_{i,j}^E \leftarrow D_{i,j}^G + 1/(p_a + 1) * D_{i,j}^I$
    else if s_i in A^s **then**
      $D_{i,j}^E \leftarrow 1/(|p_a - j| + 1)$
    else if s_j in A^s
      $D_{i,j}^E \leftarrow 1/(|p_a - i| + 1)$
    else
      $D_{i,j}^E \leftarrow D_{i,j}^I$
    ▷ The process of constructing the sentiment interaction graph
    if s_i and s_j in T and A^s **then**
      $A_{i,j}^C \leftarrow 1 + 1/(|p_a - p_o| + 1) * (1/(p_o \sum_{n=1}^{N} \frac{p_o}{p_o + 1} + 1)$
    else if s_i in A^s and s_j in A^o **then**
      $A_{i,j}^C \leftarrow 1 + 1/(|p_a - p_o| + 1) * (1/(p_o \sum_{n=1}^{N} \frac{p_o}{p_o - j} + 1))$
    else if s_j in A^s and s_i in A^o **then**
      $A_{i,j}^C \leftarrow 1 + 1/(|p_a - p_o| + 1) * (1/(p_o \sum_{n=1}^{N} \frac{p_o}{p_o - i} + 1))$
    else
      $A_{i,j}^C \leftarrow 0$
  ▷ The process of improving and enriching the sentiment interaction graph
  for $i = 1 \to n; j = 1 \to n$ **do**
    $A_{i,j}^C \leftarrow A_{j,i}^C$
return A^C

3.3. Multi-graph perception mechanism

In the process of generating sentiment interaction graphs, we construct the dependency graph, interaction graph, enhancement graph and sentiment interaction graph to capture the internal semantic correlations and sentiment interaction relations. In order to improve the semantic and syntactic dependencies in each graph, the multi-graph perception mechanism is used to capture the dependency information which cannot be captured between different graphs, and to reduce the overlap of information. Intuitively, the related entities of each word should be in different regions in a sentence, so we expect the model to learn syntactic information and dependencies by interaction between the various graphs. Meanwhile, we avoid introducing overlapping information in the process of constructing a graph. The perception mechanism is as follows:

$$R = 1/(\sum_{n=1}^{N} Regular(A^C A^O) + 1) \tag{8}$$

$$Regular(A^C||A^O) = \frac{1}{2}KL(A^C||\frac{A^C + A^O}{2}) + \frac{1}{2}KL(A^O||\frac{A^C + A^O}{2}) \tag{9}$$

where R represents the multi-graph perception, $Regular(A^C||A^O)$ penalizes bifurcation between A^O and A^C, A^O denotes the collections of dependency graphs, interaction graphs and enhancement graphs, and *KL* represents Kullback-Leibler divergence. A sentence contains four categories of graph, namely a dependency graph, an interaction graph, an enhancement graph and a sentiment interaction graph. The multi-graph perception mechanism considers the degree of similarity between these graphs. If they are similar, they tend to 0. If the difference is large, they tend to 1.

3.4. Graph convolutional network

In SPGCN, the sentiment interaction graph A^C and hidden state h_t^c are input into a GCN. Then, the GCN performs a convolution on the top of the LSTM in the form of an L-layer, i.e., H^l = H^c, to create context-aware nodes. Finally, the hidden representation of each node is updated through a graph convolution operation with a normalization factor:

$$h_i^l = Relu(\sum_{j=1}^{n} A_{i,j} W_l \; g_j^{l-1}/d_i + b_l) \tag{10}$$

where $h_i^l \in \mathbb{R}^{2d_h}$ is the hidden representation of the l-th GCN layer, $g_j^{l-1} \in \mathbb{R}^{2d_h}$ is the representation evolved from the $(l-1)$-th GCN layer, d_h is the dimensionality of a hidden state vector output by a unidirectional LSTM, $A_{i,j} \in \mathbb{R}^{n \times n}$ denotes the adjacency matrix of sentiment interactions, W_l is the weight matrix, b_l is the bias vector to be learned during training, and $d_i = \sum_{j=1}^{n} A_{i,j}$ represents the degree of the tree.

Specifically, in order to reduce the noise during the graph convolution process, we conduct a position-aware transformation before h_i^l is input into GCN:

$$g_i^l = p_i h_i^l \tag{11}$$

$$p_i = \begin{cases} 1 - |i-k|/n, & 0 < i < k+1 \\ 0, & k+1 \leq i \leq k+m \\ 1 - |k+m-i|/n, & k+m < i \leq n \end{cases} \tag{12}$$

where p_i is the position weight of the i-th token, and h_i^l denotes the hidden representation of the l-th GCN layer. After the processing of LSTM and SPGCN, we can obtain the hidden representations of the sentiment interaction graph, i.e., h^G, h^I, h^E, and h^C. After that, we combine these final representations to extract the semantic and syntactic dependencies, and adopt the multi-graph perception mechanism to alleviate the overlapping information:

$$h_i^{sp} = h_i^G + \alpha * h^I + R * (h^E + h^C) \tag{13}$$

where α is the coefficient of the interaction features. We construct a masking mechanism on top of the GCN to mask out non-aspect words. This mechanism enhances the specific aspects to perceive context through syntactic information and long-range sentiment dependencies:

$$H^L_{Mask} = \begin{cases} 0, & 0 < t < k+1 \\ h_t^{sp}, & k+1 \leq t \leq k+m \\ 0, & k+m < t \leq n \end{cases} \tag{14}$$

where h_t^{sp} is the representation of the t-th word learned by SPGCN. Then, we adopt the retrieval-based attention representation based on H^c and H^L_{mask} and formulate it as follows:

$$h^R = \sum\nolimits_{t=1}^{n} \alpha_t h_t^c \tag{15}$$

$$\alpha_t = \frac{exp\left(\beta_t\right)}{\sum_{i=1}^{n} exp\left(\beta_i\right)} \tag{16}$$

$$\beta_t = \sum_{i=1}^{n} \left(h_t^c\right)^\top h_i^{sp} = \sum_{i=k}^{k+m} \left(h_t^c\right)^\top h_i^{sp} \tag{17}$$

where h^R is the retrieval-based attention representation, α_t represents the attention weight, and β_t is the attention-aware function to obtain the semantic correlation between the aspect and context.

Finally, we input the attention representation h^R into the $softmax$ layer for aspect-based sentiment analysis:

$$y = softmax\left(W_p h^R + b_p\right) \tag{18}$$

where $y \in \mathbb{R}^{|C|}$ is the sentiment distribution prediction, $W_p \in \mathbb{R}^{2d_h \times |C|}$ and $b_p \in \mathbb{R}^{|C|}$ are the trainable parameters, and C is the dimension of the sentiment labels.

3.5. Model training

The purpose of model training is to optimize all the parameters and to minimize the loss function as far as possible. Our model is trained using cross-entropy with the L2-regularization term and formulated as follows:

$$loss = -\sum_{i}^{N} y_i \log\left(\hat{y}_i\right) + \lambda \|\theta\|^2 \tag{19}$$

where N is the number of samples in the datasets, y_i is the ground truth probability, $\hat{y_i}$ is the estimated probability of an aspect, λ is the $L2$-regularization factor, and θ represents all the trainable parameters.

4. Experiments

In this section, we first describe the experimental datasets in Section 4.1. Then, the implementation details and baseline models are described in Sections 4.2 and 4.3. To evaluate the performance of the proposed model, we adopt accuracy and the macro-averaged F1-score as the evaluation metrics, and compare it with advanced baseline models in Section 4.4. After that, we explore the influence of the number of GCN layers in SPGCN and the value of the interactive coefficient in Sections 4.5 and 4.6. In order to verify the influence of each component within SPGCN on performance, we perform ablation experiments on five publicly-available datasets in Section 4.7. Finally, we construct a visualization of the sentiment interaction graph to show how SPGCN can improve the performance in Section 4.8, and some cases are explored to analyze the limitations of current models in Section 4.9.

4.1. Experimental datasets

We conduct experiments on five publicly-available datasets: Lap14, Rest14, Rest15, Rest16 and Twitter. The Lap14 and Rest14 datasets are from the SemEval 2014 task 4[2], Rest15 is from SemEval 2015 task 12[3], Rest16 is from SemEval 2016 task 5[4], and Twitter[5] is based on *Twitter* data. The SemEval datasets consist of *Restaurant* and *Laptop* datasets. All the datasets include three sentiment polarity labels: positive, negative, and neutral. The dataset statistics are shown in Table 1.

4.2. Implementation Details

In our experiments, we apply the pretrained GloVe vectors with 300 dimensions

[2]http://alt.qcri.org/semeval2014/task4/
[3]http://alt.qcri.org/semeval2015/task12/
[4]http://alt.qcri.org/semeval2016/task5/
[5]http://goo.gl/5Enpu7

Table 1: Statistics of datasets

Datasets	Positive		Negative		Neutral	
	Train	Test	Train	Test	Train	Test
Lap14	994	341	464	169	870	128
Rest14	2164	728	807	193	637	196
Rest15	978	326	307	182	50	35
Rest16	1620	579	709	190	88	38
Twitter	1561	173	1528	169	3016	336

to initialize the word embeddings [6]. The dimension of the hidden state vectors is thus also set to 300. All the weight matrices obtain their initial values from a Xavier random uniform distribution. All the models are optimized using the Adam optimizer and the learning rate is set to 0.001. The $L2$ regularization is set to 0.00001, and the batch size is set to 16. In addition, the number of GCN layers is set to 2. In order to optimize the model training, we average the experimental results of 8 runs with random initialization.

4.3. Baseline models

We compare SPGCN with 14 state-of-the-art baselines, 8 attention-based and 6 GCN-based:

TD-LSTM [43]. TD-LSTM employs two LSTMs to model the left context and the right context for given aspects.

ATAE-LSTM [56]. ATAE-LSTM constructs the attention representation by combining aspect embedding with LSTM hidden state, and generates the input by combining aspect embedding and context embedding.

MenNet [44]. MenNet constructs a deep memory network to capture important information of context words by different attention strategies.

IAN [29]. IAN learns attention information from the contexts and aspects interactively with two LSTMs.

RAM [13]. RAM adopts a multiple-attention mechanism to capture sentiment features separated by a long distance.

[6] `https://nlp.stanford.edu/projects/glove/`

AOA [21]. AOA constructs an attention model to capture the specific representations for aspects and contexts.

TNET [25]. TNET proposes a target-specific transformation component to preserve and strengthen the informative parts of contexts.

PAM [60]. PAM adopts a novel and concise architecture using two Bidirectional GRUs along with an attention layer to classify each aspect based on its context words.

ASGCN [63]. ASGCN exploits syntactic dependency structures within a sentence and resolves the long-range multiword dependency issue for aspect-based sentiment classification.

CDT [42]. CDT exploits Bi-directional LSTMs to learn aspect and context representations, and further enhances the embedding with a graph convolutional network.

R-GAT [51]. R-GAT constructs a unified aspect-oriented dependency tree structure rooted at a target aspect by reshaping and pruning an ordinary dependency parse tree.

BiGCN [64]. BiGCN employs a global lexical graph to capture word co-occurrence information, and builds a concept hierarchy to differentiate dependency relations.

KumaGCN [11]. KumaGCN proposes gating mechanisms to dynamically combine information from word dependency graphs and latent graphs which are learned by self-attention networks.

SenticGCN [26]. SenticGCN constructs the graph neural networks by integrating the affective knowledge from SenticNet to enhance the dependency graphs of sentences.

4.4. Results and analysis

To evaluate the performance of the proposed model, we adopt Accuracy and Macro-averaged F1-score as the evaluation metrics. The Macro-F1 metric is more appropriate when the data set is unbalanced. The results are shown in Table 2. SPGCN achieves the best performance relative to the 14 baselines on all five datasets, i.e., Lap14, Rest14, Rest15, Rest16 and Twitter, as measured by accuracy. Firstly, compared with the 8 models based on LSTM and an attention mechanism, we think the reason why SPGCN performs well is that it can capture syntactic information and long-range sentiment dependencies in encoding the aspects and contextual information, which avoids mistakenly identifying irrelevant contextual words as clues for judging aspect

Table 2: Performance of SPGCN compared to 14 previous models, using five publicly-available datasets

Model		Lap14		Rest14		Rest15		Rest16		Twitter	
		Acc	Macro-F1	Acc	Macro-F1	Acc	Macro-F1	Acc	Macro-F1	Acc	Macro-F1
Attention-based	TD-LSTM	71.80	68.46	78.00	68.43	76.39	58.70	82.16	54.21	69.89	66.21
	ATAE-LSTM	68.88	63.93	78.60	67.02	78.48	62.84	83.77	61.71	70.14	66.03
	MenNet	70.64	65.17	79.16	69.53	77.89	59.52	83.04	57.91	71.48	69.90
	IAN	71.95	67.14	79.42	70.01	78.18	52.45	84.74	55.21	72.45	71.26
	RAM	74.49	71.35	80.23	70.80	79.98	60.57	83.88	62.14	-	-
	AOA	73.11	68.47	79.06	70.13	78.17	57.02	87.50	66.21	72.30	70.20
	TNET	74.95	70.16	80.77	72.03	78.19	60.67	88.72	70.16	72.57	71.13
	PAM	75.39	70.50	81.37	72.06	80.88	62.48	89.30	66.93	-	-
GCN-based	ASGCN	75.55	71.05	80.77	72.02	79.89	61.89	88.99	67.48	72.15	70.40
	CDT	77.19	72.99	82.30	74.02	-	-	85.58	69.93	74.66	**73.66**
	R-GAT	77.42	73.76	82.84	74.86	-	-	-	-	74.36	72.41
	BiGCN	74.59	71.84	81.97	73.48	81.16	64.79	88.96	70.84	74.16	73.35
	KumaGCN	76.12	72.42	81.43	73.64	80.69	65.99	89.39	73.19	72.45	70.77
	SenticGCN	76.02	72.08	81.78	73.96	80.78	64.75	89.56	71.94	-	-
Our model	SPGCN	**77.90**	**73.86**	**83.16**	**74.91**	**81.37**	**68.07**	**90.75**	**75.20**	**74.86**	72.95

sentiment. In particular, in contrast to the conventional GCN, SPGCN enables the model to capture the internal semantic correlations among aspect phrases, and contextual information is fused into syntactic dependencies to enhance the capability of context-awareness. Secondly, compared to the 6 GCN models, they merely consider the syntactic dependencies between the specific aspect and the context, which ignores the sentiment interaction relations among different aspects of sentences. SPGCN, by contrast, captures the available sentiment effects of different aspects and integrates them into adjacency graphs. In the subsequent ablation study, we will further analyze the results to verify our hypothesis.

4.5. Influence of number of GCN layers

We evaluate SPGCN on the same datasets with the number of GCN layers ranging from 1 to 9. As shown in Figs. 3 and 4, when the number of GCN layers is set to 2, SPGCN achieves the best performance in terms of Accuracy and Macro-F1.

On the one hand, with a 1-layer GCN, SPGCN's node representations cannot propagate far enough to capture the syntactic dependencies. On the other hand, when

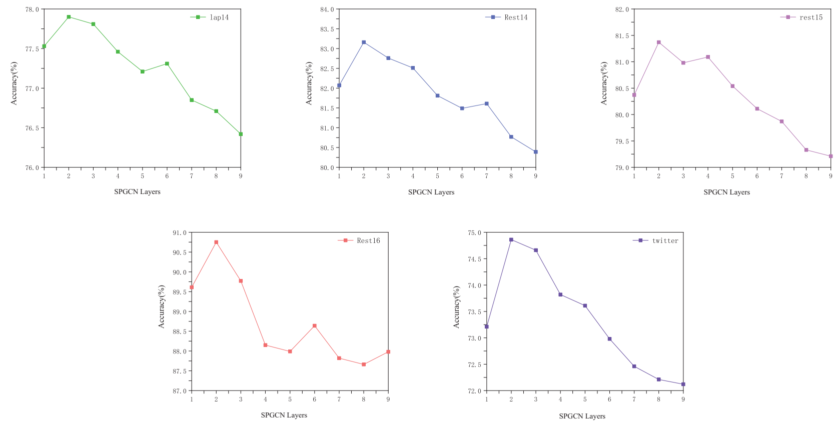

Fig. 3. Influence of the numbers of SPGCN layers on Accuracy

the number of GCN layers is excessive, we observe the phenomenon of over-smoothing, which makes the features of all nodes become more and more similar.

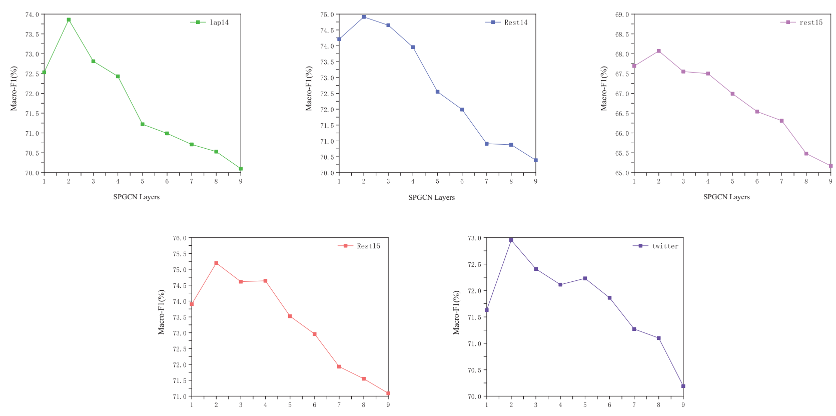

Fig. 4. Influence of the numbers of SPGCN layers on Macro-F1

4.6. Influence of interactive coefficient

To further analyze the effect of the interaction graph in SPGCN, we conduct experiments based on different values of the interactive coefficient α. Based on syntactic dependencies, the interaction graph uses the internal semantic correlations between aspect phrases to enhance aspect-centric dependencies.

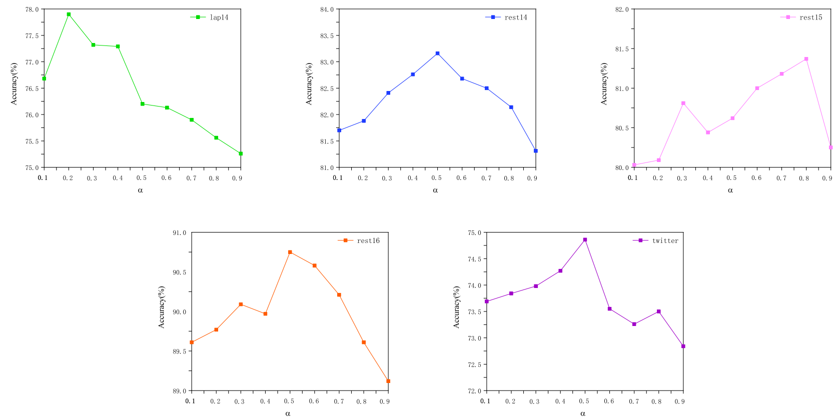

Fig. 5. Influence of the interactive coefficient α on Accuracy

As demonstrated in Figs. 5 and 6, when the coefficient α is from 0.1 to 0.2, we can observe that the performance of SPGCN on the Lap14 dataset steadily improves. When the coefficient is from 0.3 to 0.9, the performance of the model decreases. Meanwhile, as shown in Table 3, we see that the proportion of sentences which contain aspect phrases (henceforth aspect phrase ratio) is 44% for Lap14. We believe that appropriately incorporating the interaction features extracted from the interaction graph can help the model learn semantic relationships within the aspect phrase and hence improve the performance of aspect sentiment analysis. Therefore, when the coefficient α is 0.2, the model achieves the best performance. The performance changes of the Rest14, Rest16 and Twitter datasets are different from Lap14: With the first three datasets, when the coefficient α is from 0.1 to 0.5, the performance of SPGCN steadily improves, and the performance decreases when the coefficient α is from 0.6 to 0.9. Similarly, we see that

Table 3: Ratio of aspect phrases in the datasets

Datasets	Lap14	Rest14	Rest15	Rest16	Twitter
Aspect phrase ratio	44%	34%	28%	31%	67%

the aspect phrase ratios for Rest14 and Rest16 are 33% and 31%, i.e. lower values than for Lap14 (44%). So the model needs further learning in order to capture the internal semantic associations of different aspect words among aspect phrases. Significantly, the aspect phrase ratio for Twitter is 67%, i.e. much higher than Lap14, but when the coefficient α is 0.5, the model just achieves the best performance. We speculate that the reason for this is that the structure of the Twitter dataset is different from the other four datasets. There is only one aspect on Twitter and the dense aspect phrases in the learning process bring noise. Finally, the aspect phrase ratio on Rest15 is the lowest in all the datasets, only 28%, and the model achieves the best performance when the coefficient α is 0.7. Therefore, the model needs to learn more about the internal semantic correlations of aspect phrases in order to better extract features and improve the performance.

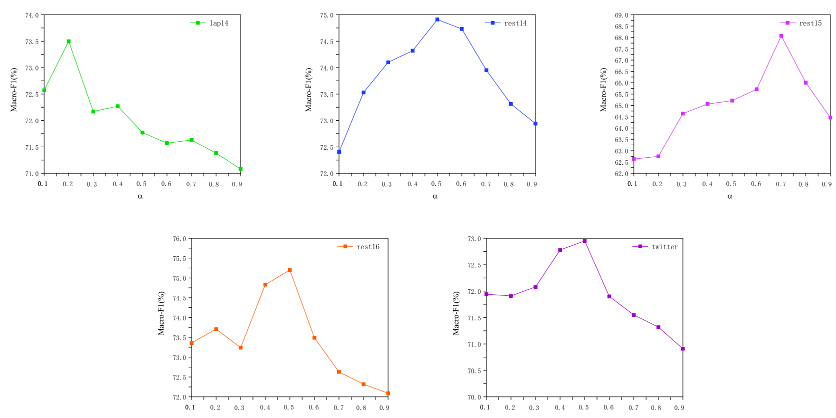

Fig. 6. Influence of the interactive coefficient α on Macro-F1

4.7. Ablation study of SPGCN

In order to verify the influence of each SPGCN component on performance, we carry out ablation experiments on the same five datasets (see Table 4). SPGCN w/o D^I means that we remove the interaction graph so that the model cannot study the semantic information of aspect phrases. We can observe the slight degradation in the performance of SPGCN on all datasets, and it shows that recognizing aspect phrases as an entirety will lose internal semantic information. SPGCN w/o D^E means that we remove the enhancement graph. Performance is further degraded compared with SPGCN w/o D^I, showing that the aspect-specific contextual information enriches the

Table 4: Ablation study on SPGCN, using the same five datasets

Model	Lap14		Rest14		Rest15		Rest16		Twitter	
	Acc	Macro-F1	Acc	Macro-F1	Acc	Macro-F1	Acc	Macro-F1	Acc	Macro-F1
SPGCN $w/o\ D^I$	76.33	72.42	82.69	74.50	80.63	64.26	90.26	74.26	73.41	72.06
SPGCN $w/o\ D^E$	76.12	72.32	82.50	72.15	80.25	63.97	89.29	72.30	72.83	71.69
SPGCN $w/o\ D^C$	75.08	71.36	82.23	72.87	79.89	63.30	89.13	71.95	72.42	70.49
SPGCN $w/o\ R$	75.86	72.18	81.78	73.15	79.36	62.74	90.09	70.65	73.12	71.02
SPGCN	**77.90**	**73.86**	**83.16**	**74.91**	**81.37**	**68.07**	**90.75**	**75.20**	**74.86**	**72.95**

dependency of specific aspect words to a certain extent. Meanwhile, we can observe that the performance decline of removing the sentiment interaction graph D^C is greater than that of the interaction graph and enhancement graph, verifying that there are intricate sentiment relationships in sentences, meaning that sentiment interaction relations among different aspects are the important improvement for aspect-based sentiment analysis. For the multi-graph perception mechanism, we believe that the model can identify the relevant entities of each word even if they are located in different areas of the sentence, so SPGCN can learn syntactic information and dependencies via interaction between the graphs. Meanwhile, we avoid introducing overlapping information in the process of constructing the graph.

4.8. Visualization of sentiment interaction graph

In order to show qualitatively how the proposed SPGCN can improve the performance of aspect sentiment analysis, we visualize in Fig. 7 the internal semantic correlations and sentiment interaction relations of an example sentence "Macbook notebooks quickly die out because of their short battery life, as well as the many unknowable background programs" (see also Fig. 2 earlier) by displaying the interaction graph and sentiment interaction graph.

We can observe in the figure that the interaction graph constructs the semantic correlations between aspect phrases. Among them, there are sentiment associations between "MacBook" and "notebooks", "battery" and "life", and "background" and "programs". SPGCN captures the sentiment influence between different words in

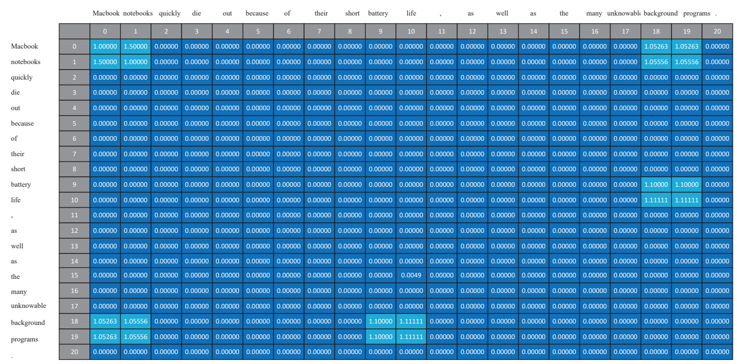

Fig. 7. Visualization of the interaction graph

the same aspect through the interaction graph, and adds semantic information to the adjacency matrix to help the model capture the contextual dependencies of aspects towards a specific aspect.

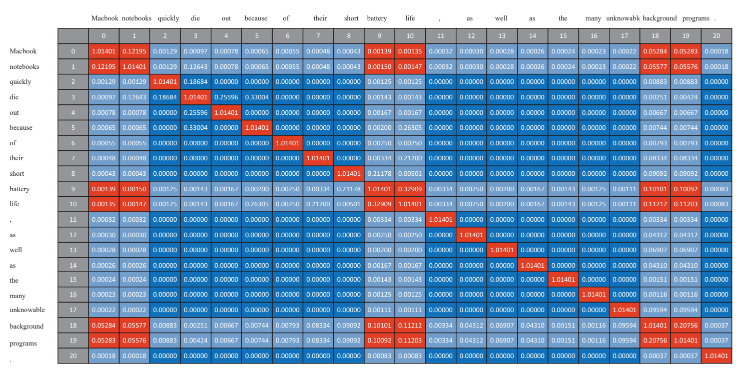

Fig. 8. Visualization of the sentiment interaction graph

As shown in Fig. 8, based on the same example "Macbook notebooks..." the sentiment interaction graph not only pays attention to the features of a node itself, but also integrates the features of other nodes that have sentiment interaction relations on this node and adds them to that node itself. So, because short battery life and unknowable background programs lead to a Macbook notebook quickly dying out, SPGCN pays more attention to the Macbook notebooks while paying attention to short battery life and unknowable background programs. Meanwhile, the words in the syntactic tree and aspect word sets also have certain influences on the specific aspect words.

4.9. Case study of SPGCN

In order to analyze the limitations of current models and verify the performance of the model, we analyzed some example cases using different models. *P*, *N* and *O* represent positive, negative and neutral sentiment respectively. Red, blue, and green are used to highlight different aspects.

Table 5: Case study of proposed SPGCN model compared with baselines

No	Review	PAM	ASGCN	SPGCN
1	Air has higher resolution but the fonts are small.	$(N_{\times}, N_{\checkmark})$	$(P_{\checkmark}, N_{\checkmark})$	$(P_{\checkmark}, N_{\checkmark})$
2	Works well, and I am extremely happy to be back to an apple OS.	$(P_{\checkmark}, P_{\checkmark})$	$(P_{\checkmark}, P_{\checkmark})$	$(P_{\checkmark}, P_{\checkmark})$
3	There was a little difficulty doing the migration as the firewire cable system can't be used with the ibook.	$(O_{\times}, N_{\times})$	$(O_{\times}, O_{\checkmark})$	$(N_{\checkmark}, O_{\checkmark})$
4	This place is really fashion but they have forgotten about the most important part of a restaurant, the food.	$(P_{\checkmark}, P_{\times})$	$(P_{\checkmark}, P_{\times})$	$(P_{\checkmark}, N_{\checkmark})$
5	Macbook notebooks quickly die out because of their short battery life, as well as the many unknowable background programs.	$(O_{\times}, N_{\checkmark}, N_{\checkmark})$	$(O_{\times}, N_{\checkmark}, N_{\checkmark})$	$(N_{\checkmark}, N_{\checkmark}, N_{\checkmark})$

As shown in Table 5, for the aspect “resolution” in the first sample, PAM, which is based on an attention mechanism, incorrectly focuses on the word “small”. ASGCN and SPGCN, on the other hand, accurately analyze the sentiment polarity of “resolution” and “fonts”. This first sample proves that models based on GCN can flexibly capture syntactic information and word dependencies. In the second sample, for aspects “firewire cable system” and “ibook”, PAM cannot recognize either the semantic dependencies of “firewire cable system” or the long-range dependencies of “ibook”. ASGCN successfully recognizes the sentiment polarity of the second aspect, but makes a mistake in the first aspect. SPGCN correctly identifies the sentiment polarity of both aspects, due to its consideration of internal semantic correlations. In the fourth sample, we can observe that the sentence implicitly describes the polarities of aspects. All three models correctly identify the sentiment polarity of aspect “place”, but PAM and ASGCN mistakenly recognize the sentiment of aspect “the food”. SPGCN captures the influences of context on specific aspects to correctly recognize the sentiments of all aspects. In the fifth sample, PAM and ASGCN successfully identify the sentiment polarity of "battery life" and "background programs", but cannot correctly identify the sentiment polarity of "Macbook notebooks". We believe that the reason for this identification error is that the previous models cannot consider the sentiment dependency relationships between aspects. SPGCN fully considers the possible sentiment interaction relations of different aspects, so as to make the model conduct more accurate aspect-based sentiment analysis.

5. Conclusion and future work

In this paper, we explored for the first time the internal semantic correlations among aspect phrases and the sentiment interaction relations among different aspects. We proposed a graph convolutional network with sentiment interaction graphs and a multi-graph perception mechanism for aspect-based sentiment analysis. The proposed model derives internal semantic information for aspect words within specific aspect phrases to enrich aspect-centric dependencies. Contextual dependencies on specific aspect words are then fused to syntactic information to enhance the capability of context-awareness. Subsequently, the sentiment interactions between different aspects are integrated into syntactic dependencies to generate the sentiment interaction graph. In addition, a multi-graph perception mechanism is constructed to determine the specific dependency information that cannot be captured between different graphs and hence to reduce the amount of overlapping information. Experimental results on five publicly-available datasets demonstrate that our proposed model outperforms state-of-the-art methods.

We have two directions of future work to improve SPGCN. The first direction is to rebuild the dependency tree. In this paper, we generate the trees using the spaCy toolkit, but this may make syntax parsing errors. Therefore, we hope to generate a new syntactic dependency tree from which to construct the adjacency matrix. The second direction is to assist sentiment dependencies by incorporating external information such as knowledge graphs. SPGCN can identify and analyze sentences with multiple semantic correlations and sentiment interaction relations. However, for sentences that implicitly express sentiment polarity, it is usually impossible to infer their sentiment from the explicit sentiment clues. A knowledge graph with reasoning ability can infer and judge implicit sentiment through its rich knowledge. So, combining these two approaches will be a focus in our future research.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant No. 61877050, by the Key R & D plan of Xianyang under Grant No. 2021ZDYF-GY-0033, and by the Program for International Science and Technology Cooperation Projects of Shaanxi Province under Grant No. 2021KW-63.

References

[1] Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R.J., 2011. Sentiment analysis of twitter data, in: Proceedings of the workshop on language in social media (LSM 2011), pp. 30–38.

[2] Al-Smadi, M., Qawasmeh, O., Al-Ayyoub, M., Jararweh, Y., Gupta, B., 2018a. Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of arabic hotels' reviews. Journal of computational science 27, 386–393.

[3] Al-Smadi, M., Qawasmeh, O., Al-Ayyoub, M., Jararweh, Y., Gupta, B., 2018b. Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of arabic hotels' reviews. Journal of computational science 27, 386–393.

[4] Bao, L., Lambert, P., Badia, T., 2019. Attention and lexicon regularized lstm for aspect-based sentiment analysis, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, pp. 253–259.

[5] Bi, J.W., Liu, Y., Fan, Z.P., 2019. Representing sentiment analysis results of online reviews using interval type-2 fuzzy numbers and its application to product ranking. Information Sciences 504, 293–307.

[6] Brychcín, T., Konkol, M., Steinberger, J., 2014. Uwb: Machine learning approach to aspect-based sentiment analysis, in: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pp. 817–822.

[7] Cai, H., Tu, Y., Zhou, X., Yu, J., Xia, R., 2020. Aspect-category based sentiment analysis with hierarchical graph convolutional network, in: Proceedings of the 28th International Conference on Computational Linguistics, pp. 833–843.

[8] Cambria, E., 2016. Affective computing and sentiment analysis. IEEE Intelligent Systems 31, 102–107.

[9] Cambria, E., Kumar, A., Al-Ayyoub, M., Howard, N., 2022a. Guest editorial: Explainable artificial intelligence for sentiment analysis. Knowledge-Based Systems 238.

[10] Cambria, E., Liu, Q., Decherchi, S., Xing, F., Kwok, K., 2022b. Senticnet 7: A commonsense-based neurosymbolic ai framework for explainable sentiment analysis, in: Proceedings of LREC.

[11] Chen, C., Teng, Z., Zhang, Y., 2020a. Inducing target-specific latent structures for aspect sentiment classification, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 5596–5607.

[12] Chen, G., Tian, Y., Song, Y., 2020b. Joint aspect extraction and sentiment analysis with directional graph convolutional networks, in: Proceedings of the 28th International Conference on Computational Linguistics, pp. 272–279.

[13] Chen, P., Sun, Z., Bing, L., Yang, W., 2017. Recurrent attention network on memory for aspect sentiment analysis, in: Proceedings of the 2017 conference on empirical methods in natural language processing, pp. 452–461.

[14] Da'u, A., Salim, N., Rabiu, I., Osman, A., 2020. Weighted aspect-based opinion mining using deep learning for recommender system. Expert Systems with Applications 140, 112871.

[15] Do, H.H., Prasad, P., Maag, A., Alsadoon, A., 2019. Deep learning for aspect-based sentiment analysis: a comparative review. Expert Systems with Applications 118, 272–299.

[16] Gu, S., Zhang, L., Hou, Y., Song, Y., 2018. A position-aware bidirectional attention network for aspect-level sentiment analysis, in: Proceedings of the 27th international conference on computational linguistics, pp. 774–784.

[17] Han, J., Cheng, B., Wang, X., 2020. Hypergraph convolutional network for multi-hop knowledge base question answering (student abstract), in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 13801–13802.

[18] Hazarika, D., Poria, S., Vij, P., Krishnamurthy, G., Cambria, E., Zimmermann, R., 2018. Modeling inter-aspect dependencies for aspect-based sentiment analysis, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pp. 266–270.

[19] He, R., Lee, W.S., Ng, H.T., Dahlmeier, D., 2017. An unsupervised neural attention model for aspect extraction, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 388–397.

[20] Huang, B., Carley, K.M., 2018. Parameterized convolutional neural networks for aspect level sentiment classification, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 1091–1096.

[21] Huang, B., Ou, Y., Carley, K.M., 2018. Aspect level sentiment classification with attention-over-attention neural networks, in: International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation, Springer. pp. 197–206.

[22] Huang, L., Ma, D., Li, S., Zhang, X., Wang, H., 2019. 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), pp. 3444–3450.

[23] Huang, Q., Wei, J., Cai, Y., Zheng, C., Chen, J., Leung, H.f., Li, Q., 2020. Aligned dual channel graph convolutional network for visual question answering, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 7166–7176.

[24] Khalil, T., El-Beltagy, S.R., 2016. Niletmrg at semeval-2016 task 5: Deep convolutional neural networks for aspect category and sentiment extraction, in: Proceedings of the 10th international workshop on semantic evaluation (SEMEVAL-2016), pp. 271–276.

[25] Li, X., Bing, L., Lam, W., Shi, B., 2018. Transformation networks for target-oriented sentiment classification, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 946–956.

[26] Liang, B., Su, H., Gui, L., Cambria, E., Xu, R., 2022. Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. Knowledge-Based Systems 235, 107643.

[27] Liang, B., Yin, R., Gui, L., Du, J., Xu, R., 2020. Jointly learning aspect-focused and inter-aspect relations with graph convolutional networks for aspect sentiment analysis, in: Proceedings of the 28th International Conference on Computational Linguistics, pp. 150–161.

[28] Liu, Q., Zhang, H., Zeng, Y., Huang, Z., Wu, Z., 2018. Content attention model for aspect based sentiment analysis, in: Proceedings of the 2018 World Wide Web Conference, pp. 1023–1032.

[29] 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.

[30] Ma, Y., Peng, H., Cambria, E., 2018a. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm, in: Proceedings of the AAAI Conference on Artificial Intelligence.

[31] Ma, Y., Peng, H., Khan, T., Cambria, E., Hussain, A., 2018b. Sentic lstm: a hybrid network for targeted aspect-based sentiment analysis. Cognitive Computation 10, 639–650.

[32] Mukherjee, A., Liu, B., 2012. Aspect extraction through semi-supervised modeling, in: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 339–348.

[33] Pavlopoulos, J., Androutsopoulos, I., 2014. Aspect term extraction for sentiment analysis: New datasets, new evaluation measures and an improved unsupervised method, in: Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM), pp. 44–52.

[34] Pecar, S., 2018. Towards opinion summarization of customer reviews, in: Proceedings of ACL 2018, Student Research Workshop, pp. 1–8.

[35] Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., Mohammad, A.S., 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.

[36] Poria, S., Cambria, E., Gelbukh, A., 2016. Aspect extraction for opinion mining with a deep convolutional neural network. Knowledge-Based Systems 108, 42–49.

[37] Ruder, S., Ghaffari, P., Breslin, J.G., 2016. A hierarchical model of reviews for aspect-based sentiment analysis, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 999–1005.

[38] Schouten, K., Frasincar, F., 2015. Survey on aspect-level sentiment analysis. IEEE Transactions on Knowledge and Data Engineering 28, 813–830.

[39] Serrano-Guerrero, J., Olivas, J.A., Romero, F.P., Herrera-Viedma, E., 2015. Sentiment analysis: A review and comparative analysis of web services. Information Sciences 311, 18–38.

[40] Song, M., Park, H., Shin, K.s., 2019. Attention-based long short-term memory network using sentiment lexicon embedding for aspect-level sentiment analysis in korean. Information Processing & Management 56, 637–653.

[41] Song, W., Wen, Z., Xiao, Z., Park, S.C., 2021. Semantics perception and refinement network for aspect-based sentiment analysis. Knowledge-Based Systems 214, 106755.

[42] Sun, C., Huang, L., Qiu, X., 2019. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence, 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), pp. 380–385.

[43] Tang, D., Qin, B., Feng, X., Liu, T., 2016a. Effective lstms for target-dependent sentiment classification, in: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pp. 3298–3307.

[44] Tang, D., Qin, B., Liu, T., 2016b. Aspect level sentiment classification with deep memory network, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 214–224.

[45] Tang, H., Ji, D., Li, C., Zhou, Q., 2020. Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 6578–6588.

[46] Tay, Y., Tuan, L.A., Hui, S.C., 2018. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis, in: Proceedings of the AAAI conference on artificial intelligence.

[47] Tian, Y., Chen, G., Song, Y., 2021. 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, pp. 2910–2922.

[48] Toh, Z., Su, J., 2015. Nlangp: Supervised machine learning system for aspect category classification and opinion target extraction, in: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pp. 496–501.

[49] Valdivia, A., Luzón, M.V., Cambria, E., Herrera, F., 2018. Consensus vote models for detecting and filtering neutrality in sentiment analysis. Information Fusion 44, 126–135.

[50] Wang, D., Zhu, S., Li, T., 2013. Sumview: A web-based engine for summarizing product reviews and customer opinions. Expert Systems with Applications 40, 27–33.

[51] Wang, K., Shen, W., Yang, Y., Quan, X., Wang, R., 2020a. Relational graph attention network for aspect-based sentiment analysis, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 3229–3238.

[52] Wang, W., Pan, S.J., Dahlmeier, D., Xiao, X., 2016a. Recursive neural conditional random fields for aspect-based sentiment analysis, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 616–626.

[53] Wang, W., Yang, N., Wei, F., Chang, B., Zhou, M., 2017. Gated self-matching networks for reading comprehension and question answering, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 189–198.

[54] Wang, X., Tang, M., Yang, T., Wang, Z., 2021a. A novel network with multiple attention mechanisms for aspect-level sentiment analysis. Knowledge-Based Systems 227, 107196.

[55] Wang, Y., Chen, Q., Shen, J., Hou, B., Ahmed, M., Li, Z., 2021b. Aspect-level sentiment analysis based on gradual machine learning. Knowledge-Based Systems 212, 106509.

[56] Wang, Y., Huang, M., Zhu, X., Zhao, L., 2016b. Attention-based lstm for aspect-level sentiment classification, in: Proceedings of the 2016 conference on empirical methods in natural language processing, pp. 606–615.

[57] Wang, Z., Ho, S.B., Cambria, E., 2020b. Multi-level fine-scaled sentiment sensing with ambivalence handling. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 28, 683–697.

[58] Xing, B., Liao, L., Song, D., Wang, J., Zhang, F., Wang, Z., Huang, H., 2019. Earlier attention? aspect-aware lstm for aspect-based sentiment analysis. arXiv preprint arXiv:1905.07719 .

[59] 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.

[60] Yadav, R.K., Jiao, L., Goodwin, M., Granmo, O.C., 2021. Positionless aspect based sentiment analysis using attention mechanism. Knowledge-Based Systems 226, 107136.

[61] Yang, H., Zeng, B., Yang, J., Song, Y., Xu, R., 2021. A multi-task learning model for chinese-oriented aspect polarity classification and aspect term extraction. Neurocomputing 419, 344–356.

[62] Yao, L., Mao, C., Luo, Y., 2019. Graph convolutional networks for text classification, in: Proceedings of the AAAI conference on artificial intelligence, pp. 7370–7377.

[63] Zhang, C., Li, Q., Song, D., 2019. Aspect-based sentiment classification with aspect-specific graph convolutional networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 4568–4578.

[64] Zhang, M., Qian, T., 2020. 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), pp. 3540–3549.

[65] Zhou, J., Huang, J.X., Hu, Q.V., He, L., 2020. Sk-gcn: modeling syntax and knowledge via graph convolutional network for aspect-level sentiment classification. Knowledge-Based Systems 205, 106292.