Journal of Intelligent Information Systems (2023) 60:97–117
https://doi.org/10.1007/s10844-022-00729-1

Exploring rich structure information for aspect-based sentiment classification

Ling Zhu[1] · Xiaofei Zhu[1] · Jiafeng Guo[2] · Stefan Dietze[3,4]

Received: 2 March 2022 / Revised: 13 July 2022 / Accepted: 13 July 2022 /
Published online: 30 July 2022

Abstract

Graph Convolutional Network (GCN) for aspect-based sentiment classification has attracted a lot of attention recently due to their promising performance in handling complex structure information. However, previous methods based on GCN focused mainly on examining the structure of syntactic dependency relationships, which were subject to the noise and sparsity problem. Furthermore, these methods tend to focus on one kind of structural information (namely syntactic dependency) while ignoring many other kinds of rich structures between words. To tackle these problems, we propose a novel GCN based model, named Structure-Enhanced Dual-Channel Graph Convolutional Network (SEDC-GCN). Specifically, we first exploit the rich structure information by constructing a text sequence graph and an enhanced dependency graph, then design a dual-channel graph encoder to model the structure information from the two graphs. After that, we propose two kinds of aspect-specific attention, i.e., aspect-specific semantic attention and aspect-specific structure attention, to learn sentence representation from two different perspectives, i.e., the semantic perspective based on the text encoder, and the structure perspective based on the dual-channel graph encoder. Finally, we merge the sentence representations from the above two perspectives and obtain the final sentence representation. We experimentally validate our proposed model SEDC-GCN by comparing with seven strong baseline methods. In terms of the metric accuracy, SEDC-GCN achieves performance gains of 74.42%, 77.74%, 83.30%, 81.73% and 90.75% on TWITTER, LAPTOP, REST14, REST15, and REST16, respectively, which are 0.35%, 4.22%, 1.62%, 0.70% and 2.01% better than the best performing baseline BiGCN. Similar performance improvements are also observed in terms of the metric macro-averaged F1 score. The ablation study further demonstrates the effectiveness of each component of SEDC-GCN.

Keywords Aspect-based sentiment classification · Graph convolutional networks · Attention mechanism · Sentiment analysis

✉ Xiaofei Zhu
zxf@cqut.edu.cn

Extended author information available on the last page of the article

1 Introduction

Aspect-based sentiment analysis (ABSA) is a fine-grained sentiment analysis task (Singh and Singh 2021; Zhu et al. 2021; Berka 2020), which aims at identifying the aspect term and their sentiment polarities. Figure 1 shows an example of aspect-based sentiment analysis with multiple sentiment polarities, and in the review "*The appetizers are excellent, while the service is quite slow.*", there are two aspects (i.e., "*appetizers*" and "*service*") and the sentiment labels corresponding to the two aspects within this review are "positive" and "negative", respectively. ABSA can be divided into two strongly coupled subtasks including opinion aspect extraction (OAE) and aspect-based sentiment classification (ABSC). The former focuses on extracting the explicit aspects which consist of the single word aspect and the compound aspect, while the latter sheds light on identifying sentiment polarity of a specified aspect in a review. In this paper, we focus on the task of ABSC, which is challenging due to the fact that a review may contain various sentiments for different aspects.

Early studies on aspect-based sentiment classification mainly rely on employing machine learning classification methods and labor-intensive hand-crafted features as their key features (Kiritchenko et al. 2014; Wagner et al. 2014). In recent years, many neural network models such as Recurrent Neural Networks (RNNs) (Liu and Zhang 2017; Wang et al. 2016), which is augmented by the attention mechanism (Bahdanau et al. 2015), have been utilized for the task of aspect-based sentiment classification. The attention mechanism can facilitate the identification of potentially relevant words with respect to the target aspect (Yang et al. 2017). Some researchers also propose to utilize convolutional neural networks (CNNs) (Xue and Li 2018; Kalchbrenner et al. 2014; Conneau et al. 2017) to capture multi-word phrase information for aspect-based sentiment classification. More recently, many research efforts have been devoted to exploit the syntactic dependency tree (Zhang et al. 2019; Liang et al. 2019), and apply a graph convolutional network (GCN) (Yao et al. 2019; Linmei et al. 2019) to capture long-distant relationships among words.

Although state-of-the-art performance has been achieved, existing GCN-based methods mainly focus on capturing words' distant relationship based on the structure of the syntactic dependency tree, which would lead to inferior performance due to the sparsity issue. In addition, these methods usually model one kind of structure information (i.e. syntactic dependency structure) while neglect other kinds of rich structure information between words, such as the consecutive structure of words within a time window, or the co-occurrence structure between words in the entire corpus.

In this paper, we propose a novel GCN based model, called Structure-Enhanced Dual-Channel Graph Convolutional Network (SEDC-GCN), for dealing with the task of aspect-based sentiment classification. More precisely, we first exploit the rich structure information by constructing a text sequence graph and an enhanced dependency graph. The former is utilized to capture the sequential structure information among words, i.e., words within a small window in a text sequence are connected and yields the sequential structure. The latter is leveraged to incorporate the syntactic dependency structure information among

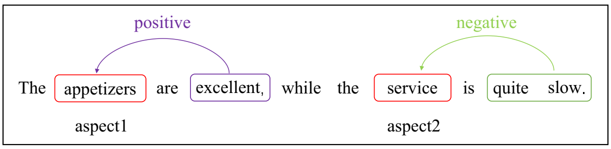


Fig. 1 An example of aspect-based sentiment classification with multiple sentiment polarities

words, i.e., each text sequence corresponds to a syntactic dependency tree, where the collection of such dependencies in the sequence yields the syntactic dependency structure.

To deal with the sparsity issue of the dependency structure, we further enrich the syntactic dependency structure by introducing high correlated relationships between words based on their global co-occurrence information. Then a dual-channel graph encoder is designed to model the two kinds of structure information. Note that in the dual-channel graph encoder, a CoGCN (Coordinate Graph Convolution Network) module is developed to seamlessly integrate these two kinds of structure information in a mutual reinforcement manner. In particular, CoGCN consists of a multi-layer Co-Attention and a gate layer, where the former is developed to ensure the collaboration of different types of information to be performed at different levels and the latter is designed to merge information from both perspective (i.e., the text sequence graph and the enhanced dependency graph). After that, we design an aspect-specific structure attention module to obtain sentence representation. Similarly, we also leverage a text encoder to obtain word representation sequence from the semantic perspective, and design an aspect-specific semantic attention module to obtain sentence representation on top of the text encoder. Finally, we merge the sentence representations from the above two perspectives with the concatenation operation and obtain the final sentence representation. The main contribution of this work are summarized as follows:

- We exploit the rich structure information by constructing a text sequence graph and an enhanced dependency graph.
- We propose two kinds of aspect-specific attention, i.e., aspect-specific semantic attention and aspect-specific structure attention, to obtain sentence representation from two different perspectives, i.e., the semantic perspective and the structure perspective.
- We design a dual-channel graph encoder to effectively model the two kinds of rich structure information.
- We conduct extensive experiments on five widely used datasets, and the results demonstrate that our proposed approach has overwhelming superiority over the state-of-the-art baseline methods in terms of both metrics, i.e., Accuracy and F1 score.

2 Related work

Aspect-based sentiment classification is a common task in natural language processing. Most of the early works use basic machine learning classification methods with labor-intensive hand-crafted features as their key features (Kiritchenko et al. 2014; Wagner et al. 2014). In recent years, many neural network models such as Recurrent Neural Networks (RNNs) (Liu and Zhang 2017; Wang et al. 2016) have been utilized for the task of aspect-based sentiment classification. Tang et al. (2015) model the relatedness of a target word with its context words based on the long short-term memory (LSTM) (Hochreiter and Schmidhuber 1997). Tang et al. (2016) attempt to explore the attention mechanism to explicitly capture the importance of context words by developing a deep memory network. Wang et al. (2016) develop an aspect-to sentence attention mechanism to enforce the model to attend to the important context words. Chen et al. (2017) model the sentiment of the phrase-like word sequence based on position-weighted memories and a multiple-attention mechanism. Ma et al. (2017) learn context by using attention mechanisms associated with targets to gain relevant information about context, and they then model the target based on the information determined by the attention mechanisms. Some researchers also propose to adopt convolutional neural networks (CNNs) (Xue and Li 2018; Kalchbrenner et al. 2014; Conneau et al. 2017) for aspect-based sentiment classification, which has been shown that competitive performance can be achieved by capturing multi-word phrases. Xue et al. (2018) employ convolutional neural networks and gating mechanisms. It has two separate convolutional layers on the top of the embedding layer, and a gating unit is then used to combine the output of the two convolutional layers.

More recently, inspired by the success of applying graph neural network (GNN) in a wide variety of tasks, including node classification, text classification, some researchers have recently tried to employ GNN to improve the performance of aspect-based sentiment classification. For example, Zhang et al. (2019) applied a graph convolutional network (GCN) to capture long-range relationships between words. Specifically, they take the output of LSTM as the initial representation of word nodes in a sentence, then apply a two-layer graph convolutional network to obtain syntactic structure features, and retain only specific aspects of features through a masking mechanism, and finally obtain sentence representation by aspect-aware attention. Sun et al. (2019) leverage the Bi-LSTM to account for contextual information between successive words, and then exploit a GCN to model the dependencies along the syntactic paths of the dependency tree. Huang et al. (2020) incorporate external pre-training knowledge by BERT to model the interaction between the context and aspect words, and utilize the graph attention network on the dependency tree structure to model the long-distance dependency. Liang et al. (2022) propose a graph convolutional network based on SenticNet, which exploits the contextual affective knowledge of a sentence with respect to the target aspect. Xiao et al. (2022) develop a relational graph attention network (RGAT) based on a part-of-speech guided syntactic dependency graph, which is constructed to capture the information in the syntactic dependency trees.

Although these GNN-based methods above have achieved promising results and become the state-of-the-arts, they still have some limitations. First, these methods mainly model the word distant relationship based on the structure of the syntactic dependency tree, which would suffer from the sparsity issue. Given an example sentence "It fucking harry potter weekend annoying!", a colloquial sentence in twitter. Since this sentence has no explicit syntactic structure, it will inevitably lead to a sparse extracted syntactic dependency structure. Second, these works only focus on modeling one kind of structure information (i.e. syntactic dependency structure) while ignore other kinds of rich structure information between words, such as the consecutive structure of words within a sentence, the co-occurrence structure between words in the corpus. Differ from existing research works, our proposed approach attempts to simultaneously model both the text sequence structure and the enhanced dependency structure via a dual-channel graph encoder. It is worth noting that we expand the syntactic dependency tree structure by further accounting for the global word co-occurrence information in order to alleviate the sparsity issue.

3 Our approach

We assume that there is a sentence $S = (w_1, \cdots, w_{\tau+1}, \cdots, w_{\tau+m}, \cdots, w_n)$ consisting of n-words, where $(w_{\tau+1}, \cdots, w_{\tau+m})$ represents the aspect. The framework of our proposed model SEDC-GCN is shown in Fig. 2, which mainly consists of three components: 1) *Text Encoder*. It aims to model the contextual information of the text; 2) *Dual-Channel Graph Encoder*. This component is designed to capture the structure information from

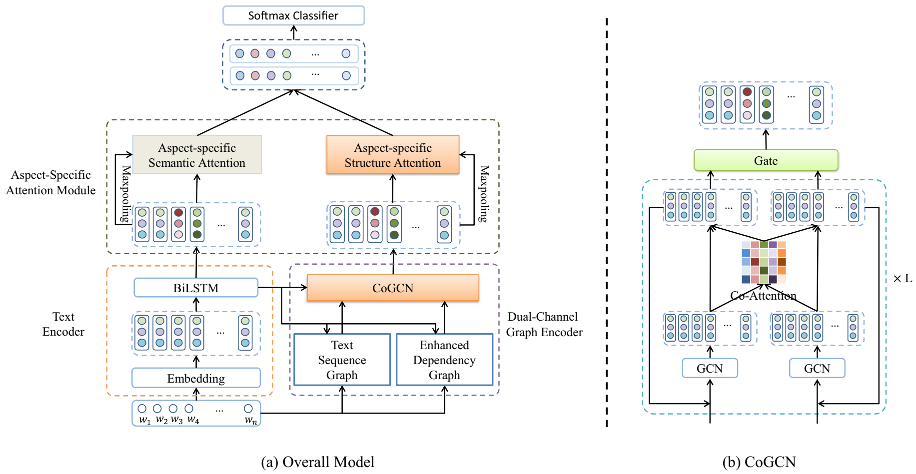


Fig. 2 The framework of the proposed approach SEDC-GCN

two different channels, i.e., the text sequence structure channel and the enhanced dependency structure channel. A CoGCN module is employed to fuse information from the two channels; 3) *Aspect-Specific Attention Module*. It is utilized to learn aspect-specific sentence representations based on the outputs of the text encoder and the dual-channel graph encoder.

3.1 Text encoder

The text encoder first embeds the sentence S to a word embedding sequence $(e_1, \cdots, e_{\tau+1}, \cdots, e_{\tau+m}, \cdots, e_n) \in \mathbb{R}^{n \times d_e}$ with a pre-trained matrix of word embeddings $M \in \mathbb{R}^{|V| \times d_e}$, where $|V|$ denotes the vocabulary size and d_e is the dimension of the word embedding. Intuitively, as the objective of aspect-based sentiment classification is to identify the sentiment polarity of the target aspect, it should focus on words relating to the aspect while alleviate the influence of other less relevant words. Given the sentence "The appetizers are excellent, while the service is quite slow." and the target aspect "appetizers", it should pay more attention to these close words (e.g., "excellent") instead of these distant ones (e.g., "quite slow"). Therefore, words that are closer to the aspect in position should contribute more in judging the sentiment of the aspect (Gu et al. 2018), and we introduce the absolute distance from each context word w_t to the corresponding aspect, and get a position sequence $(a_1, \cdots, a_{\tau+1}, \cdots, a_{\tau+m}, \cdots, a_n)$ for S. Then a position embedding lookup table with random initialization $E_p \in \mathbb{R}^{n \times d_a}$ is utilized to map the position sequence to a position embedding sequence $(p_1, \cdots, p_{\tau+1}, \cdots, p_{\tau+m}, \cdots, p_n) \in \mathbb{R}^{n \times d_a}$, where d_a is the dimension of the position embedding. For each word w_i, its embedding e'_i is represented as the concatenation of its word embedding e_i and position embedding p_i, i.e., $e'_i = [e_i;p_i] \in \mathbb{R}^{d_w}$, where $d_w = d_e + d_a$, and [;] denotes the concatenation operation.

To capture the contextual representation of each word w_i in S, we employ the bidirectional long short-term memory (BiLSTM) (Zhou et al. 2016) to learn the word embedding $h_i \in \mathbb{R}^{2d_w}$ for w_i as follows:

$$\overrightarrow{h}_i = LSTM(\overrightarrow{h}_{i-1}, e'_i) \tag{1}$$

$$\overleftarrow{h}_i = LSTM(\overleftarrow{h}_{i-1}, e'_i) \tag{2}$$

where $h_i = [\overrightarrow{h}_i;\overleftarrow{h}_i]$, $\overrightarrow{h}_i \in \mathbb{R}^{d_w}$ and $\overleftarrow{h}_i \in \mathbb{R}^{d_w}$ are the forward and backward representations, respectively. Finally, we get the output of the text encoder $H = (h_1, \cdots, h_{\tau+1}, \cdots, h_{\tau+m}, \cdots, h_n) \in \mathbb{R}^{n\times 2d_w}$.

3.2 Dual-channel graph encoder

After obtaining the contextual representations from the text encoder, we develop a dual-channel graph encoder to model the structure information from two different perspectives, i.e., the text sequence structure channel and the enhanced dependency structure channel. We leverage a CoGCN module to seamlessly integrate these two kinds of structure information in a mutual reinforcement manner. As demonstrated in Fig. 2, dual-channel graph encoder mainly consists of two graph (i.e., the text sequence graph and the enhanced dependency graph) construction modules and a CoGCN module.

3.2.1 Text sequence graph (SeqGCN)

Inspired by Huang et al. (2019), we build a text sequence graph for each text sequence, where word nodes within a small window in the sequence are connected, and the representations of nodes will be updated in the text sequence graph through a GCN, i.e., a node aggregates information from its neighboring nodes to update its representation. In particular, to construct a graph for a given text, we first treat all words that appear in the text as graph nodes, and connect two words if they are in the same window. Note that we set a threshold p to gather information for different window size in the text. Formally, the sequence graph of the text S is defined as:

$$V = \{h_i | i \in [1, n]\} \tag{3}$$

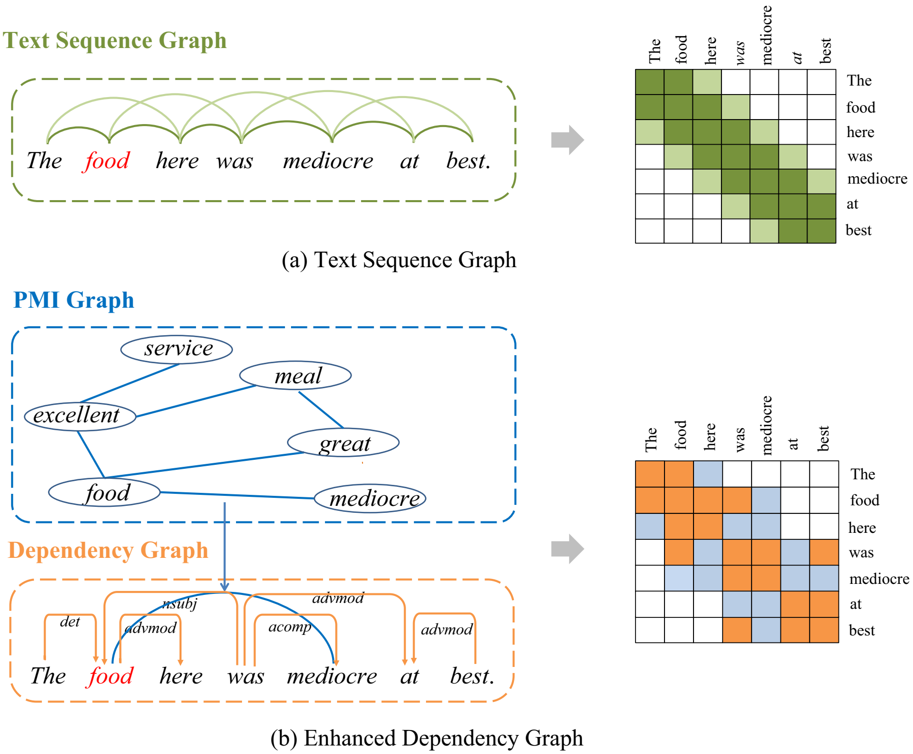
$$E = \{e_{ij} | i \in [1, n]; j \in [i-p, i+p]\} \tag{4}$$

where h_i denotes the representaion of the i-th word, V and E are the node set and edge set of the graph. Figure 3a shows an example of the construction process of the text sequence graph. A node i and node j's edge weight is expressed in the following form:

$$A^s_{ij} = \begin{cases} 1 & i \in [1, n], j \in [i-p, i+p] \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

As suggested in Zhang et al. (2019), self-loops are added to each node and the activations are normalized in the graph convolution prior to converting it into linearity as follows:

$$\hat{h}^{s,(l)}_i = g\left(\sum_{j=1}^{n} \tilde{A}^s_{ij} W^{s,(l)} h^{s,(l-1)}_j / d^s_i + b^{s,(l)}\right) \tag{6}$$



(a) Text Sequence Graph

(b) Enhanced Dependency Graph

Fig. 3 Construction processes of the two kinds of graphs. The top part corresponds to the text sequence graph, where we set p=2 for each node. The bottom part corresponds to the enhanced dependency graph, which relies on the syntactic dependency structure and the global word co-occurrence structure

$$h_i^{s,(l)} = Norm(h_i^{s,(l-1)} + \hat{h}_i^{s,(l)}) \tag{7}$$

where $h_j^{s,(l-1)}$ is the learned representation of the j-th node in the $(l-1)^{th}$ layer ($h_j^{s,(0)} = h_j$), $g(\cdot)$ is a nonlinear function, e.g., ReLU. $Norm(\cdot)$ is a normalization layer, and $\tilde{A}^s = A^s + I$ where I is the $n \times n$ identity matrix, $d_i^s = \sum_{j=1}^{n} \tilde{A}_{ij}^s$ denotes the degree of the i-th node, $W^{s,(l)}$ and $b^{s,(l)}$ are layer-specific trainable parameters, l is the layer number and i is the target node for aggregation. By applying GCN on the text sequence graph, we obtain the output representation in the l-th layer $H^{s,(l)} = (h_1^{s,(l)}, \cdots, h_{\tau+1}^{s,(l)}, \cdots, h_{\tau+m}^{s,(l)}, \cdots, h_n^{s,(l)})$.

3.2.2 Enhanced dependency graph (EdepGCN)

We also incorporate the syntactic information by modeling the syntactic dependency tree in order to capture distant relationship between words. Existing approaches (Zhang et al. 2019; Hou et al. 2021) model the word distant relationship only based on the structure of the syntactic dependency tree, which would suffer from the sparsity issue. To address this issue, we take into account the global word co-occurrence information, e.g., point-wise mutual information (PMI), to obtain word pairs with high correlation (Yao et al. 2019). Different from conventional methods (Zhang et al. 2019) which directly utilize the syntactic dependency tree, we further expand the syntactic dependency tree by adding edges with high correlated relationships, and obtain the enhanced dependency graph. Formally, the PMI value of the word pair (w_i, w_j) is calculated as:

$$PMI(i,j) = \log \frac{p(i,j)}{p(i)p(j)} \tag{8}$$

$$p(i,j) = \frac{\#W(i,j)}{\#W} \tag{9}$$

$$p(i) = \frac{\#W(i)}{\#W} \tag{10}$$

where $\#W(i)$ is the number of sliding windows containing the word w_i in the corpus, $\#W(i,j)$ is the amount of sliding windows that contain both words w_i and w_j, and $\#W$ is the number of sliding windows total in the corpus. If the PMI values of two words are positive , then they have edges. Figure 3b demonstrates an example of the construction process of the enhanced dependency graph.

It is worth noting that the weight of edges in the syntactic dependency tree is equal (e.g., 1 by default), while the PMI value is in a large range (e.g., $[0, +\infty)$). If we directly incorporate the PMI values as the weight of edges, the influence of these syntactic edges will be largely weakened. To deal with this issue, we assign an edge weight for each of the PMI edge rather than using the original PMI value as the edge weight. A node i and node j's edge weight is expressed in the following form:

$$A_{ij}^{dp} = \begin{cases} 1 & PMI(i,j) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

$$A_{ij}^{dd} = \begin{cases} 1 & w_i, w_j \text{ with a dependency edge} \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

As there are two types of edges (i.e., PMI edge and syntactic edge) in the enhanced dependency graph, we adopt a heterogeneous graph based graph convolution networks (Zhang and Qian 2020) to aggregate information from different types of edges. In this case, each node first aggregates information from the same type of edges, then concatenates the corresponding representations of each edge type as a new representation of the node:

$$\hat{h}_i^{dp,(l)} = \sigma(\sum_{j=1}^{n} \tilde{A}_{ij}^{dp} W^{d,(l)} h_j^{d,(l-1)} / d_i^{dp}) \tag{13}$$

$$\hat{h}_i^{dd,(l)} = \sigma(\sum_{j=1}^{n} \tilde{A}_{ij}^{dd} W^{d,(l)} h_j^{d,(l-1)} / d_i^{dd}) \tag{14}$$

$$h_i^{d,(l)} = Norm(h_i^{d,(l-1)} + [\hat{h}_i^{dp,(l)}; \hat{h}_i^{dd,(l)}] + b^{d,(l)}) \tag{15}$$

where $h_j^{d,(l-1)}$ is the learned representation of the j-th node in the $(l-1)^{th}$ layer ($h_j^{d,(0)} = h_j$), and $W^{d,(l)}$ is the weight matrix in the l^{th}-layer. $\tilde{A}^{dp} = A^{dp} + I$, where A^{dp} denotes the adjacency matrix corresponding to the PMI dependency of the graph and I being the $n \times n$

identity matrix. $d_i^{dp} = \sum_{j=1}^{n} \tilde{A}_{ij}^{dp}$ denotes the degree of the i-th node. $\tilde{A}^{dd} = A^{dd} + I$, where A^{dd} denotes the adjacency matrix corresponding to the syntactic dependency of the graph and I being the $n \times n$ identity matrix, $d_i^{dd} = \sum_{j=1}^{n} \tilde{A}_{ij}^{dd}$ denotes the degree of the i^{th} node. $b^{d,(l)}$ is a bias term. By applying GCN on the enhanced dependency graph, we obtain the output representation in the l-th layer $H^{d,(l)} = (h_1^{d,(l)}, \cdots, h_{\tau+1}^{d,(l)}, \cdots, h_{\tau+m}^{d,(l)}, \cdots, h_n^{d,(l)})$.

3.2.3 CoGCN

We further leverage information learnt from the text sequence graph and enhanced dependency graph to get better representation for the sentence S. To the end, we develop a CoGCN module which includes a multi-layer Co-Attention and a gate layer. The former is used to ensure the collaboration of different types of information to be performed at different levels, and the latter is utilized to merge information from both perspectives. Figure 2b illustrates the architecture of the CoGCN module.

We leverage the output of text encoder to initialize the node representations of both the text sequence graph and the enhanced dependency graph, i.e., $H^{s,(0)} = (h_1, \cdots, h_{\tau+1}, \cdots, h_{\tau+m}, \cdots, h_n)$ and $H^{d,(0)} = (h_1, \cdots, h_{\tau+1}, \cdots, h_{\tau+m}, \cdots, h_n)$. Then we employ the collaborative attention (Ma et al. 2019) to integrate information from both graph channels as follows:

$$A_1 = softmax(H^{s,(l)} W_1 {H^{d,(l)}}^T) \tag{16}$$

$$A_2 = softmax(H^{d,(l)} W_2 {H^{s,(l)}}^T) \tag{17}$$

where $H^{s,(l)} \in \mathbb{R}^{n \times 2d_w}$ and $H^{d,(l)} \in \mathbb{R}^{n \times 2d_w}$ are the learned representation from the text sequence graph and the enhanced dependency graph in the l-th Co-Attention layer, respectively. $W_1 \in \mathbb{R}^{2d_w \times 2d_w}$, $W_2 \in \mathbb{R}^{2d_w \times 2d_w}$ are all trainable parameters, $A_1 \in \mathbb{R}^{n \times n}$ and $A_2 \in \mathbb{R}^{n \times n}$ are the temporary alignment matrices projecting from $H^{d,(l)}$ to $H^{s,(l)}$ and $H^{s,(l)}$ to $H^{d,(l)}$, respectively. Then we have the updated representations $H^{s,(l)}, H^{d,(l)}$ as follows:

$$H^{s,(l)} = A_2 H^{d,(l)} \tag{18}$$

$$H^{d,(l)} = A_1 H^{s,(l)} \tag{19}$$

$$H^{s,(l)}, H^{d,(l)} = CoAttention(H^{s,(l)}, H^{d,(l)}) \tag{20}$$

where CoAttention represents Equations (16)-(19).

We use the gating mechanism to fuse the two learnt representations at the last layer L, and get a new feature representation with complementary enhancements as follows:

$$g = \sigma(W_g[h_i^{s,(L)}; h_i^{d,(L)}]) \tag{21}$$

$$h^g = g h_i^{s,(L)} + (1 - g) h_i^{d,(L)} \tag{22}$$

where σ denotes the sigmoid function, L is the number of layers of the CoGCN, $W_g \in \mathbb{R}^{2d_w \times 4d_w}$ are learnable parameters. Finally, the structure representation of the text is $H^g = (h_1^g, \cdots, h_{\tau+1}^g, \cdots, h_{\tau+m}^g, \cdots, h_n^g) \in \mathbb{R}^{n \times 2d_w}$.

3.3 Aspect-specific attention module

In this subsection, we introduce the process of obtaining sentence representations from two different perspectives, i.e., the semantic perspective based on the text encoder, and the structure perspective based on the dual-channel graph encoder. In particular, we apply two kinds of aspect-specific attention, i.e., *Aspect-specific Semantic Attention* and *Aspect-specific Structure Attention*, to obtain sentence representation from different perspectives.

Aspect-specific Semantic Attention: From the text encoder module, we get semantic representation of the sentence $H = (h_1, \cdots, h_{\tau+1}, \cdots, h_{\tau+m}, \cdots, h_n)$. We mask non-aspect words and keep aspect word unchanged in H, and get a zero-masked representation $H_{mask\_a} = (0, \cdots, h_{\tau+1}, \cdots, h_{\tau+m}, \cdots, 0)$. Then we utilize the max-pooling operation to get the aspect representation $h_a \in \mathbb{R}^{2d_w}$.

Finally, we retrieve the important features that are semantically related to the aspect, and set the retrieval-based attention weights (Zhang et al. 2019) for each word. The final semantic representation z_{sem} for the sentence is formulated as:

$$e_t = h_a^T h_t \tag{23}$$

$$a_t = \frac{exp(e_t)}{\sum_{i=1}^{n} exp(e_i)} \tag{24}$$

$$z_{sem} = \sum_{t=1}^{n} a_t h_t \tag{25}$$

where a_t is the attention score of the t-th word with respect to the aspect.

Aspect-specific Structure Attention: From the Dual-Channel Graph Encoder, we obtain the structure representation of the text $H^g = (h_1^g, \cdots, h_{\tau+1}^g, \cdots, h_{\tau+m}^g, \cdots, h_n^g)$. The contextual information related to the aspect is retrieved from a structure perspective. Similar to the aspect-specific semantic attention module, we get a zero-masked representation $H_{mask\_b} = (0, \cdots, h_{\tau+1}^g, \cdots, h_{\tau+m}^g, \cdots, 0)$ and then apply the max-pooling operation on H_{mask_b} to obtain the aspect representation $h_b \in \mathbb{R}^{2d_w}$. The final structure representation z_{stru} for sentence is formulated as follows:

$$e_t = h_b^T h_t^g \tag{26}$$

$$a_t = \frac{exp(e_t)}{\sum_{i=1}^{n} exp(e_i)} \tag{27}$$

$$z_{stru} = \sum_{t=1}^{n} a_t h_t^g \tag{28}$$

where a_t is the attention score of the t-th respect to the aspect.

After we get the sentence representations from both semantic and structure perspectives, we merge them with the concatenation operation and obtain the final aspect-specific sentence representation r as follows:

$$r = [z_{sem}; z_{stru}] \tag{29}$$

3.4 Final representation and model training

We feed r to a fully connected layer and a softmax layer to generate the probability distribution over sentiment labels $y \in \mathbb{R}^{d_y}$, as follows:

$$y = softmax(W_y r + b_y) \tag{30}$$

where d_y is the number of labels, and $W_y \in \mathbb{R}^{d_y \times 4d_w}$ and $b_y \in \mathbb{R}^{d_y}$ represents trainable weights and bias.

To train the classifiers, the objective is to minimize the cross-entropy loss between the predicted probability and the ground truth:

$$\mathscr{L} = -\sum_{d \in \mathscr{Y}_D} \sum_{k=1}^{d_y} Y_{dk} \log y_{dk} + \lambda ||\Theta||_2 \tag{31}$$

where $\mathscr{Y}_D$ represents the set of sentences with labels, Y is the ground-truth label matrix, y_{dk} is the predicted probability of the sentence d to the k-th label, λ is the coefficient of L_2 regularization, and Θ denotes all parameters. Figure 4 illustrates the flow diagram of the proposed methodology.

4 Experiment

4.1 Datasets

Our experiments are conducted on five benchmark datasets, including Twitter, LAPTOP, REST14, REST15, and REST16. Specifically, The dataset Twitter is originally built from Dong et al. (2014). Both LAPTOP and REST14 are constructed from SemEval 2014 task 4 (Pontiki et al. 2014). REST15 and REST16 are constructed from SemEval 2015 task 12 (Pontiki et al. 2015) and SemEval 2016 task 5 (Pontiki et al. 2016), respectively. Following previous studies (Tang et al. 2016; Zhang et al. 2019; Zhang and Qian 2020), we remove the samples with conflicting polarities and those without explicit aspects in the sentences. The statistics of datasets are demonstrated in Table 1.

4.2 Compared methods

Our proposed model (SEDC-GCN) was compared to the following methods:

- LSTM (Tang et al. 2016): It proposes a target-dependent LSTMs to model the interaction between the target and the context words.
- MemNet (Tang et al. 2016): MemNet leverages deep memory network for aspect level sentiment classification. It utilizes multi-hop attention layers on the context word embeddings for sentence representation.
- AOA (Huang et al. 2018): It captures the interaction between aspect words and contextual words by employing an attention-over-attention module (Cui et al. 2017).
- IAN (Ma et al. 2017): It proposes an interactive attention network to interactively learn the attention scores in the context and the target, and generates representations of the target and the context, respectively.

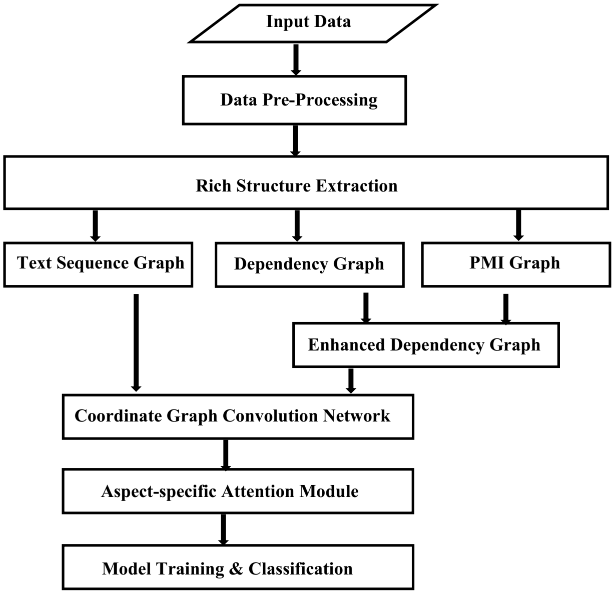


Fig. 4 Illustration of the proposed methodology

Table 1 Statistics of the datasets

Dataset		#Positive	#Neutral	#Negative
TWITTER	Train	1561	3127	1560
	Test	173	346	173
LAPTOP	Train	994	464	870
	Test	341	169	128
REST14	Train	2164	637	807
	Test	728	196	196
REST15	Train	912	36	256
	Test	326	34	182
REST16	Train	1240	69	439
	Test	469	30	117

– TNet-LF (Li et al. 2018): This model applies a target-specific transformation component to better integrate target information into the word representations.
– ASGCN (Zhang et al. 2019): It apply a multi-layered graph convolution structure within dependency graphs to capture long-distance syntactical relationships among words.
– BiGCN (Zhang and Qian 2020): It uses hierarchical graph structure to integrate word co-occurrence information and dependency type information.

4.3 Experimental settings

For SEDC-GCN, we initialize the word embeddings with 300-dimensional GloVe vectors (Pennington et al. 2014). The dimensionality of the position (i.e., the relative position of each word in a sentence with respect to the aspect) embeddings is set to 30. The threshold p for constructing the text sequence graph is set to 2, the sliding window size of the enhanced dependency graph is set to 3, and the number of CoGCN layers is set to 3. The dropout rate for the input word embeddings of BiLSTM is set to 0.4. We use Adam as the optimizer with a learning rate of 0.003. The L_2 regularization coefficient is 10^{-5} and the batch size is 16. We implement SEDC-GCN using PyTorch and conduct the dependency parsing using Spacy. All the experiments are conducted on the hardware with Intel Core CPU I7-9700K 3.6 GHz and NVIDIA GeForce GTX 2080TI. Our code and dataset are available at https://github.com/julin1991/SEDC-GCN.

4.4 Main results

As can be seen in Table 2, and we can observe that our proposed method SEDC-GCN achieves the best results on all datasets, which proves that it is superior. In particular, the baseline method LSTM obtains the worst results on five datasets. MemNet achieves a better performance than LSTM because it utilizes the multi-hops attention on the contextual word embeddings for sentence representation. AOA and IAN both rely on the attention mechanism to capture the interaction information between context and opinion target and demonstrate a better performance than MemNet. However,TNet-LF proposes to exploit a aspect-specific transformation component to better apply aspect information into the word representations. Obviously, it outperforms the previously mentioned baseline models. ASGCN uses syntactic dependency graphs to learn relational representations of distant words. The best performing baseline method BiGCN leverages the hierarchical graph structure to integrate word co-occurrence information and dependency tree information. Compared with all baselines, our proposed approach SEDC-GCN achieves the best performance. In particular, in terms of the metric accuracy, SEDC-GCN achieves performance gains of 74.42%, 77.74%, 83.30%, 81.73% and 90.75% on TWITTER, LAPTOP, REST14, REST15, and REST16, respectively, which are 0.35%, 4.22%, 1.62%, 0.70% and 2.01% better than the best performing baseline BiGCN. Similar performance improvements are also observed in terms of the metric macro-averaged F1 score. The main reason is that we develop a dual-channel graph encoder to effectively capture the rich structure information from two different perspectives.

Table 2 Performance comparison of all methods in terms of accuracy (ACC) and Macro-averaged F1 score (F1). The best performance on each dataset are in bold

Model	TWITTER		LAPTOP		REST14		REST15		REST16	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
LSTM	69.56	67.70	69.28	63.09	78.13	67.47	77.37	55.17	86.80	63.88
MemNet	71.48	69.90	70.64	65.17	79.61	69.64	77.31	58.28	85.44	65.99
AOA	72.30	70.20	72.62	67.52	79.97	70.42	78.17	57.02	87.50	66.21
IAN	72.50	70.81	72.05	67.38	79.26	70.09	78.54	52.65	84.74	55.21
TNet-LF	72.98	71.43	74.61	70.14	80.42	71.03	78.47	59.47	89.07	70.43
ASGCN	72.15	70.40	75.55	71.05	80.77	72.02	79.89	61.89	88.99	67.48
BiGCN	74.16	73.35	74.59	71.84	81.97	73.48	81.16	64.79	88.96	70.84
SEDC-GCN	**74.42**	**73.37**	**77.74**	**74.68**	**83.30**	**77.51**	**81.73**	**66.23**	**90.75**	**73.84**

4.5 Ablation study

In this Section, we perform an ablation study to investigate how different components in our proposed model contribute to the overall performance. We compare the complete model SEDC-GCN with its four variants:

– SEDC-GCN (w/o SeqGCN): We remove the text sequence graph from the dual-channel graph encoder, where the text sequence graph is leveraged to learn representations through neighbor words within a reasonably small window.
– SEDC-GCN (w/o EdepGCN): We discard the enhanced dependency graph which explores the syntactic information extracted from a dependency tree. Note that modeling the enhanced dependency graph is useful to capture distant relationship between words.
– SEDC-GCN (w/o PMI): We remove the PMI edges from the enhanced dependency graph, where the PMI edge is leveraged to capture the global word co-occurrence information.
– SEDC-GCN (w/o Dual.): This variant does not consider the dual-channel graph encoder which models the structure information from two different perspectives, i.e., the text sequence structure channel and the enhanced dependency structure channel.
– SEDC-GCN (w/o Atten.): We replace the aspect-specific structure attention with a simple retrieval-based attention (Zhang et al. 2019).

The results of the ablation study are reported in Table 3. First, we observe that the removal of text sequence graph from the dual-channel graph encoder, i.e., SEDC-GCN (w/o SeqGCN), leads to large performance degradation on all datasets. It validates the effectiveness of capturing text sequence information through neighbor words within a reasonably small window. Similarly, the removal of the enhanced dependency graph, i.e., SEDC-GCN (w/o EdepGCN), results in a significant drop in performance, which reveals the importance of modeling the enhanced dependency graph. The removal of PMI edges from the enhanced dependency graph, i.e., SEDC-GCN (w/o PMI), leads to a substantial performance degrades on all datasets, which indicates the effectiveness of expanding the syntactic dependency tree with PMI edges. In addition, if we remove the dual-channel graph encoder, i.e., SEDC-GCN (w/o Dual.), performance will be affected, especially on the datasets LAPTOP, REST14 and REST15. This is reasonable as the dual-channel graph encoder exploits structure information from both the text sequence graph and the enhanced dependency graph, and the removal of the dual-channel graph encoder will cause more information loss as compared with SEDC-GCN (w/o SeqGCN) and SEDC-GCN (w/o EdepGCN). It also shows that the text sequence graph and the enhanced dependency graph are complementary to some extent when they are explored in the the dual-channel graph encoder. At last, the removal of aspect-specific structure attention, i.e., SEDC-GCN (w/o Atten.), results in a substantial drop in performance on all datasets, indicating the importance of the use of the aspect-specific structure attention module.

4.6 Impact of the number of layers

To investigate the impact of the number of layers in the dual-channel graph encoder, we study the performance of SEDC-GCN with various number of layers ranging from 1 to 8.

Table 3 Experiments on the ablation study on five datasets. The best results on each dataset are in bold

Ablation	TWITTER		LAPTOP		REST14		REST15		REST16	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
SEDC-GCN	**74.42**	**73.37**	**77.74**	**74.68**	**83.30**	**77.51**	**81.73**	66.23	**90.75**	73.84
SEDC-GCN (w/o SeqGCN)	73.84	72.53	77.43	73.88	82.77	76.07	80.44	**67.20**	88.80	73.18
SEDC-GCN (w/o EdepGCN)	74.13	72.55	76.96	73.10	83.21	76.84	81.37	64.40	89.61	72.85
SEDC-GCN (w/o PMI)	73.99	72.76	76.80	72.84	82.14	74.77	80.26	64.93	87.99	71.18
SEDC-GCN (w/o Dual.)	74.28	72.65	75.71	72.02	82.86	75.27	80.07	63.95	89.29	**74.29**
SEDC-GCN (w/o Atten.)	73.84	71.91	76.33	71.79	82.50	75.50	81.00	66.36	88.96	70.98

The results are demonstrated in Fig. 5. We can see that the performance on dataset TWITTER first increases with the increase of the number of layers, reaching the peak at layer 3 in terms of both metrics. After that, it starts to decline gradually when we further raise the number of layers. Similar results are observed on other datasets. This is mainly because if we choose a small number of layers, the structure information from both channels (i.e., the text sequence channel and the enhanced dependency channel) can not be well explored. On the contrary, when we employ a large number of layers, it may lead to less distinguishable node representations and suffer from the over-smoothing issue (Wang et al. 2021). It demonstrates that our model can achieve a promising performance with a relatively small number of layers.

4.7 Impact of the sliding window size

The sliding window size is leveraged to calculate word co-occurrence in the construction process of the enhanced dependency graph, i.e., if two words appear in a sliding window, then they are considered as co-occurrence. A larger sliding window size indicates that the two words would be considered as correlated even they are distant from each other in a sequence. Figure 6 shows the performance of SEDC-GCN on all five datasets with the

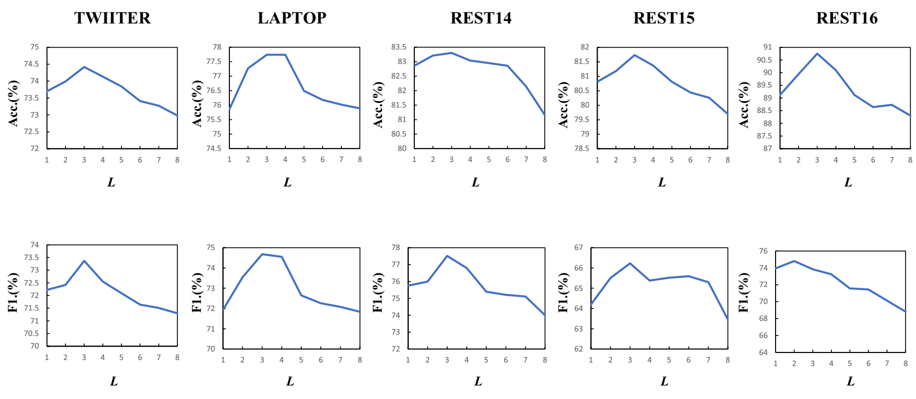


Fig. 5 The impact of the number of layers (L). The performance of SEDC-GCN with respect to different number of CoGCN Layer are reported

window size varying from 1 to 8. From Fig. 6, we can observe that the sliding window size affects the performance of SEDC-GCN in both accuracy and F1.

The changing of accuracy and F1 with different sliding window size share a similar trend on all five datasets. Specifically, with the increase of the sliding window size, the performance of SEDC-GCN first rises up until it reaches the peak when the window size equals to 3. When we keep increasing the sliding window size, it starts to decline gradually. This changing trend is reasonable as when the sliding window size is small, it could not encode enough correlation information between words, resulting in inferior performance. When the sliding window size is set to a large value, it may introduce more irrelevant correlation information, making it less effective for the task.

4.8 Performance analysis of multiple aspects

In order to further analyze the performance improvement of our proposed model SE-GCN for multi-aspects sentences, similar to Zhu et al. (2021), we also divide the test set into three groups. Figure 7 shows the results of SEDC-GCN and two best performing baseline methods on four datasets. From Fig. 7, we can observe that on the Single-Aspect & Single-Polarity category our proposed model SEDC-GCN achieves a higher accuracy than both ASGCN and BiGCN on the REST14 and REST15 datasets and a comparable performance on the two remaining datasets (i.e., LAPTOP and REST16). When considering the metric F1, our method is consistently superior to both ASGCN and BiGCN on all four datasets. The results show that SEDC-GCN can achieve a better performance when the task is simple where only each sentences only contains a single aspect.

When on the category of the Multi-Aspect & Single-Polarity, our method demonstrates a competing performance compared to the two competing baselines on all datasets in terms of both accuracy and F1. It is interesting that all three methods have obtained a better performance on the Multi-Aspect & Single-Polarity category than the other categories (i.e., Single-Aspect & Single-Polarity and Multi-Aspect & Multi-Polarity). This may be because that the sentiment classification task becomes easier on the Multi-Aspect & Single-Polarity category. When a sentence contains multiple aspects with the same polarity, the sentiment consistency between these aspects will help each other to correctly identify the sentiment polarity of each aspect.

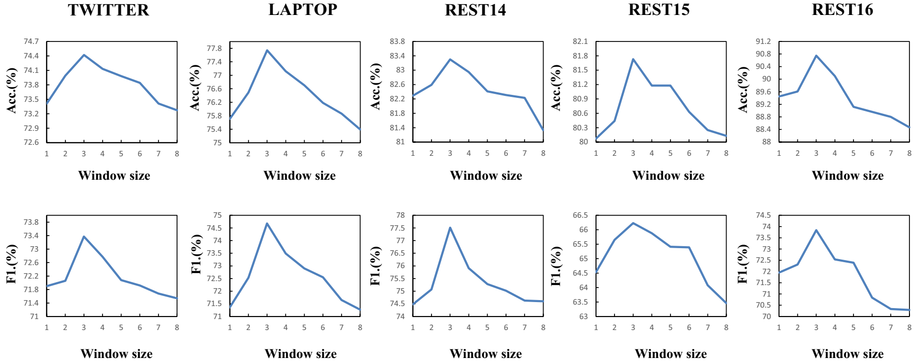


Fig. 6 The impact of the sliding window size. The performance of SEDC-GCN with respect to different sliding window size are reported

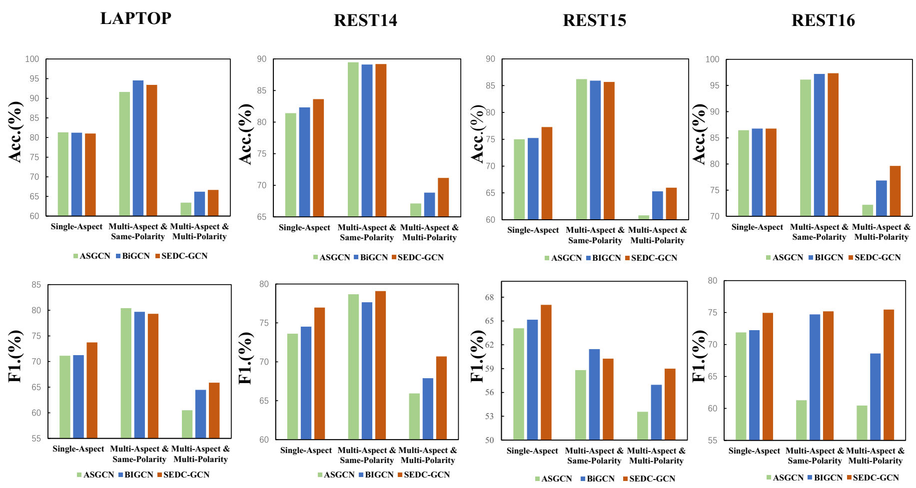


Fig. 7 Comparison of the performance of the three models in different aspect numbers

On the contrary, when the polarities of multiple aspects are different, i.e., Multi-Aspect & Multi-Polarity, the task will become more complicate. That is why all methods perform inferior on the Multi-Aspect & Multi-Polarity category as compared with the other two categories. It is worth noting that our method performs considerably better than ASGCN and BiGCN on the most difficult category in terms of both metrics. It verifies the capability of our method in modeling the corresponding information of each aspect.

4.9 Case study

In this section, we present a case study with several randomly sampled cases. Specifically, we visualize the attention scores generated by our proposed approach SEDC-GCN and two best performing baselines in Table 4. The color scale of the background indicates the attention scores of words in each sample, where a darker color corresponds to a higher attention score.

For the aspect “chinese style indian food” in the first sample, ASGCN makes a wrong prediction as it is prone to attend to the opinion word “not” which is correlated to another aspect “place” in the sample. While both BiGCN and our proposed model SEDC-GCN effectively attend to the corresponding opinion words of the aspect. Considering the second sample, both baseline methods ASGCN and BiGCN attend to improper opinion words like “bigger”, and incorrectly prediction the sentiment of the aspect “cd/dvd drive” as “negative”. In contrast, SEDC-GCN mainly attends to the aspect words themselves and assigns the correct label “neutral” to the given aspect. The last sample “a beautiful atmosphere, perfect for drinks and appetizers” which contains two aspects (i.e., “atmosphere” and “drinks”) with different sentiment attitudes. All three methods can correctly identify the sentiment of the aspect “atmosphere” via placing more attention to opinion words “beautiful” and “perfect”. However, considering the aspect “drinks”, both ASGCN and BiGCN make incorrectly classification as they attend to the opinion words “perfect”, which are the opinions towards the aspect “atmosphere” while not “drinks”. Our method SEDC-GCN assigns less attention to these unrelated opinion words like “beautiful” and “perfect” and correctly identifies the sentiment label of the aspect “drinks” as “neutral”. This result is

Table 4 Case Study. Visualization of attention scores from ASGCN, BiGCN, and SEDC-GCN on testing examples, along with their corresponding predictions and ground truth labels. The darker background color indicates a higher attention score, and the notations $\surd$ and $\times$ indicate correct and incorrect predictions, respectively

Model	Aspect	Attention visualization	Prediction	Label
ASGCN	chinese style indian food	not a very fancy place but very good chinese style indian food .	negative×	positive
	cd/dvd drive	needs a cd/dvd drive and a bigger power switch .	negative×	neutral
	atmosphere	a beautiful atmosphere , perfect for drinks and/or appetizers .	positive√	positive
	drinks	a beautiful atmosphere , perfect for drinks and/or appetizers .	positive×	neutral
BiGCN	chinese style indian food	not a very fancy place but very good chinese style indian food.	positive√	positive
	cd/dvd drive	needs a cd/dvd drive and a bigger power switch .	negative×	neutral
	atmosphere	a beautiful atmosphere , perfect for drinks and/or appetizers .	positive√	positive
	drinks	a beautiful atmosphere , perfect for drinks and/or appetizers .	positive×	neutral
SEDC-GCN	chinese style indian food	not a very fancy place but very good chinese style indian food.	positive√	positive
	cd/dvd drive	needs a cd/dvd drive and a bigger power switch.	neutral√	neutral
	atmosphere	a beautiful atmosphere , perfect for drinks and/or appetizers .	positive√	positive
	drinks	a beautiful atmosphere , perfect for drinks and/or appetizers .	neutral√	neutral

also consistent with the experimental analysis of the different number of aspects and polarities where our method performs consistently better than ASGCN and BiGCN when the polarities of multiple aspects are different. Through the above comparison, we can see that our method SEDC-GCN can correctly attend to aspect-specific opinion words via effectively modeling the rich structure information from different channels .

5 Conclusion

In this paper, we propose a novel method, named SEDC-GCN, for aspect-based sentiment classification. Specifically, we develop a dual-channel graph encoder to model the structure information from two different perspectives, i.e., the text sequence structure channel and the enhanced dependency structure channel. Then, we obtain sentence representations from two different perspectives, i.e., the semantic perspective based on the text encoder, and the structure perspective based on the dual-channel graph encoder. Experimental results on five benchmark datasets show that the proposed SEDC-GCN can perform better than state-of-the-art baseline methods. Compared to the best performing baseline BiGCN, SEDC-GCN achieves relative performance improvements of 0.35%, 4.22%, 1.62%, 0.70% and 2.01% in terms of accuracy on TWITTER, LAPTOP, REST14, REST15, and REST16, respectively. Similar performance improvements are observed in terms of the metric macro-averaged F1 score. In addition, we verify the effectiveness of each component in our proposed method SEDC-GCN, and the results show that each component plays an important role in SEDC-GCN. The impact of the number of layers and the sliding window size are also investigated, and the results demonstrate that on most of the datasets our model obtains the best performance when both the number of layers and the sliding window size equal to 3 in terms of both metrics.

It is worth noting that the main contribution of this work is to verify the effectiveness of exploiting the rich structure information by constructing a text sequence graph and an enhanced dependency graph, which is a general strategy and can be easily applied to these

BERT-based approaches. In the future work, we plan to investigate its effectiveness with regard to these BERT-based frameworks.

Acknowledgements This work was supported by the National Natural Science Foundation of China [grant number 62141201]; and the Federal Ministry of Education and Research [grant number 01IS21086].

Data Availability Statement Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

References

Bahdanau, D., Cho, K., Bengio, Y. (2015) Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*, https://doi.org/10.48550/arXiv.1409.0473

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

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), https://doi.org/10.18653/v1/d17-1047

Conneau, A., Schwenk, H., Barrault, L., LeCun, Y. (2017) Very deep convolutional networks for text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 1107–1116), https://doi.org/10.18653/v1/e17-1104

Cui, Y., Chen, Z., Wei, S., Wang, S., Liu, T., Hu, G. (2017) Attention-over-attention neural networks for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics* (pp. 593–602), https://doi.org/10.18653/v1/P17-1055

Dong L, Wei F, Tan C, Tang D, Zhou M, Xu K (2014) Adaptive recursive neural network for target-dependent twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics* (pp. 49–54), https://doi.org/10.3115/v1/p14-2009

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), https://aclanthology.org/C18-1066.

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

Hou, X., Qi, P., Wang, G., Ying, R., Huang, J., He, X., Zhou, B. (2021) Graph ensemble learning over multiple dependency trees for aspect-level sentiment classification. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 2884–2894), https://doi.org/10.18653/v1/2021.naacl-main.229

Huang, B., Ou, Y., Carley, K.M. (2018) Aspect level sentiment classification with attention-over-attention neural networks. In *Proceedings of the 11th International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation* (pp 197–206), https://doi.org/10.1007/978-3-319-93372-6_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* (pp 3442–3448), https://doi.org/10.18653/v1/D19-1345

Huang, L., Sun, X., Li, S., Zhang, L., Wang, H. (2020) Syntax-aware graph attention network for aspect-level sentiment classification. In *Proceedings of the 28th International Conference on Computational Linguistics* (pp 799–810), https://doi.org/10.18653/v1/2020.coling-main.69

Kalchbrenner, N., Grefenstette, E., Blunsom, P. (2014) A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics* (pp. 655–665), https://doi.org/10.3115/v1/p14-1062

Kiritchenko, S., Zhu, X., Cherry, C., Mohammad, S. (2014) Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th International Workshop on Semantic Evaluation* (pp. 437–442), https://doi.org/10.3115/v1/s14-2076

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* (pp. 946–956), https://doi.org/10.18653/v1/P18-1087

Liang, B., Du, J., Xu, R., Li, B., Huang, H. (2019) Context-aware embedding for targeted aspect-based sentiment analysis. In *Proceedings of the 57th Conference of the Association for Computational Linguistics* (pp 4678–4683), https://doi.org/10.18653/v1/p19-1462

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. https://doi.org/10.1016/j.knosys.2021.107643

Linmei, H., Yang, T., Shi, C., Ji, H., Li, X. (2019) Heterogeneous graph attention networks for semi-supervised short text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing* (pp 4823–4832), https://doi.org/10.18653/v1/D19-1488

Liu, J., Zhang, Y. (2017) Attention modeling for targeted sentiment. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 572–577), https://doi.org/10.18653/v1/e17-2091

Ma, D., Li, S., Zhang, X., Wang, H. (2017) Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence* (pp. 4068–4074), https://doi.org/10.24963/ijcai.2017/568

Ma, Q., Yu, L., Tian, S., Chen, E., & Ng, W. W. Y. (2019). Global-local mutual attention model for text classification. *IEEE ACM Trans Audio Speech Lang Process, 27*(12), 2127–2139. https://doi.org/10.1109/TASLP.2019.2942160

Pennington, J., Socher, R., Manning, C.D. (2014) Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* (pp. 1532–1543), https://doi.org/10.3115/v1/d14-1162

Pontiki, M., Galanis, D., Pavlopoulos, J., et al. (2014) Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation* (pp. 27–35), https://doi.org/10.3115/v1/s14-2004

Pontiki, M., Galanis, D., Papageorgiou, H., Manandhar, S., Androutsopoulos, I. (2015) Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation* (pp. 486–495), https://doi.org/10.18653/v1/s15-2082

Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., Al-Smadi, M., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O., et al. (2016) Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation* (pp. 19–30), https://doi.org/10.18653/v1/s16-1055

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

Sun, K., Zhang, R., Mensah, S., Mao, Y., Liu, X. (2019) Aspect-level sentiment analysis via convolution over dependency tree. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing* (pp. 5678–5687), https://doi.org/10.18653/v1/D19-1569

Tang, D., Qin, B., Feng, X., & Liu, T. (2015). *Target-dependent sentiment classification with long short term memory.*https://doi.org/10.48550/arXiv.1512.01100

Tang, D., Qin, B., Feng, X., Liu, T. (2016a) Effective lstms for target-dependent sentiment classification. In *Proceedings of the 25th International Conference on Computational Linguistics* (pp. 3298–3307), https://doi.org/10.48550/arXiv.1512.01100

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), https://doi.org/10.18653/v1/d16-1021

Wagner, J., Arora, P., Cortes, S., Barman, U., Bogdanova, D., Foster, J., Tounsi, L. (2014) DCU: aspect-based polarity classification for semeval task 4. In *Proceedings of the 8th International Workshop on Semantic Evaluation* (pp. 223–229), https://doi.org/10.3115/v1/s14-2036

Wang, G., Ying, R., Huang, J., Leskovec, J. (2021) Multi-hop attention graph neural networks. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence* (pp. 3089–3096), https://doi.org/10.24963/ijcai.2021/425

Wang, Y., Huang, M., Zhu, X., Zhao, L. (2016) Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (pp. 606–615), https://doi.org/10.18653/v1/d16-1058

Xiao, L., Xue, Y., Wang, H., Hu, X., Gu, D., & Zhu, Y. (2022). Exploring fine-grained syntactic information for aspect-based sentiment classification with dual graph neural networks. *Neurocomputing, 471*, 48–59. https://doi.org/10.1016/j.neucom.2021.10.091

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* (pp. 2514–2523), https://doi.org/10.18653/v1/P18-1234

Yang, M., Tu, W., Wang, J., Xu, F., Chen, X. (2017) Attention based LSTM for target dependent sentiment classification. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence* (pp. 5013–5014), https://dl.acm.org/doi/10.5555/3297863.3297977.

Yao, L., Mao, C., Luo, Y. (2019) Graph convolutional networks for text classification. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence* (pp. 7370–7377), https://doi.org/10.1609/aaai.v33i01.33017370

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* (pp. 4567–4577), https://doi.org/10.18653/v1/D19-1464

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* (pp. 3540–3549), https://doi.org/10.18653/v1/2020.emnlp-main.286

Zhou, P., Qi, Z., Zheng, S., Xu, J., Bao, H., Xu, B. (2016) Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling. In *Proceedings of the 26th International Conference on Computational Linguistics* (pp. 3485–3495), https://doi.org/10.48550/arXiv.1611.06639

Zhu, X., Zhu, L., Guo, J., Liang, S., & Dietze, S. (2021). Gl-gcn: global and local dependency guided graph convolutional networks for aspect-based sentiment classification. *Expert Systems with Applications, 186,* 115712. https://doi.org/10.1016/j.eswa.2021.115712.

Authors and Affiliations

Ling Zhu[1] · Xiaofei Zhu[1] · Jiafeng Guo[2] · Stefan Dietze[3,4]

Ling Zhu
zhuling@2019.cqut.edu.cn

Jiafeng Guo
guojiafeng@ict.ac.cn

Stefan Dietze
stefan.dietze@gesis.org

[1] College of Computer Science and Engineering, Chongqing University of Technology, 400054 Chongqing, China

[2] Institute of Computing Technology, Chinese Academy of Sciences, 100190 Beijing, China

[3] Institute of Computer Science, Heinrich-Heine-University Düsseldorf, 40225 Düsseldorf, Germany

[4] Knowledge Technologies for the Social Sciences, Leibniz Institute for the Social Sciences, 50667 Cologne, Germany