Aspect-Based Sentiment Analysis via Virtual Node Augmented Graph Convolutional Networks

Runzhong Xu[✉]

University of Nottingham, Nottingham NG7 2RD, UK
xrunzhong@gmail.com

Abstract. Aspect-based sentiment analysis (ABSA) refers to a fine-grained sentiment analysis task aimed at detecting sentiment polarity towards a given aspect. Recently, graph convolutional networks (GCN) integrated with dependency trees have achieved related appealing results in ABSA. Nevertheless, most existing models fail to preserve the information of the whole graph although global information can often significantly improve their performance. To address this problem, a novel virtual node augmented graph convolutional network (ViGCN) is proposed to further enhance the performance of GCNs in the ABSA task by adding a virtual node to the graph. The virtual node can connect to all the nodes in the graph to aggregate global information from the entire graph and then propagate it to each node. In particular, we construct edges between the virtual node and other nodes based on affective commonsense knowledge from SenticNet and the semantic-relative distances between contextual words and the aspect, effectively enhancing the collected global information towards the given aspect. Extensive experiments on three benchmark datasets illustrate that the ViGCN model can beat state-of-the-art models, proving its effectiveness.

Keywords: Sentiment analysis · Opinion mining · Aspect-based sentiment analysis · Graph neural network

1 Introduction

For the past few years, aspect-based sentiment analysis (ABSA) has become a popular research field in natural language processing [5,25]. Different from sentence-level sentiment analysis, ABSA is a fine-grained task that aims at inferring the sentiment polarity (e.g., positive, negative, neural) of one specific aspect despite possible multiple aspects in a sentence. For example, in Fig. 1, the corresponding sentiment polarities of the two aspects "works" and "apple OS" are both positive.

It is a key challenge for ABSA to learn the critical sentiment information concerning the given aspect from the sentence [11,21,25]. In early works, attention mechanism-based deep learning models are a promising paradigm due to their
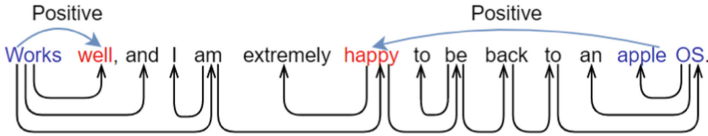
Fig. 1. An example sentence. The words in blue and red are aspects and opinions, respectively. The arrows above manifest the correspondence between aspects and opinions. The arrows below suggest dependencies between words. (Color figure online)

ability to pay attention to important parts of a sentence regarding a specific aspect [5,21]. However, these models lack a mechanism to account for sentiment dependencies between long-range words and may focus on irrelevant sentiment information to predict aspect sentiment. Recent research focuses on developing graph convolutional networks (GCN) over syntactic dependency trees [5,19,25]. Dependency trees can clarify the connections between contextual words and aspect words in a sentence. For example, in Fig. 1, there is a syntactic dependency relationship between the aspect word "Works" and its corresponding opinion word "well" as well as the other two words "am" and "and" in the sentence.

How to leverage the global structure of the graph to improve the model performance is a widely studied problem in the field of graph neural networks [1,3,12–14,23]. Most GCNs for the ABSA task, however, lack a mechanism to effectively capture the global information of the graph. To our knowledge, the GCN models in ABSA can only afford 2–3 layers [4,11,19,22,25], meaning that each node in these models can only collect local information from neighborhood nodes 2–3 hops away from it based on the message-passing scheme [14]. If a GCN model goes deeper so that each node can have a larger receptive field of the graph and learn more global information, the vanishing gradient problem will make the model unstable [11]. The nodes in the GCNs often fail to capture the critical sentiment clues due to the limitation of the receptive field. For example, in Fig. 1, the nodes representing the aspect "apple OS" are 4–5 hops away from the nodes representing the corresponding opinion word "happy". Although the opinion nodes contain significant sentiment information for determining the sentiment polarity of the aspect, this information cannot be transmitted to the aspect nodes.

To tackle the challenges mentioned above, a novel model, the virtual node augmented graph convolutional network (ViGCN), is proposed in this paper, whose architecture is shown in Fig. 2. In ViGCN, an artificial virtual node is added to the graph over the dependency tree and connected to all the real nodes to give them a global receptive field. Real nodes refer to all the nodes in the graph before adding the virtual node. The virtual node was originally proposed to represent the entire graph [9]. Recent research finds that it can be used for graph augmentation [23], because it can aggregate global information from the whole graph and propagate aggregated information to each real node. Considering that the graphs in ABSA are generated from sentences [25], the global information gathered from these graphs contains the sentiment expression of the sentences,

which is crucial for the model to predict the sentiment polarities of aspects. Moreover, under the inspiration of SenticNet [2] and previously successful LCF-BERT model [24], weighted edges between the virtual node and real nodes are established based on affective commonsense knowledge and the semantic-relative distances between contextual words and the given aspect. With this approach, the virtual node can focus more on the context containing critical sentiment information in a sentence, making the preserved global information better reflect the emotional expression of the sentence towards the given aspect.
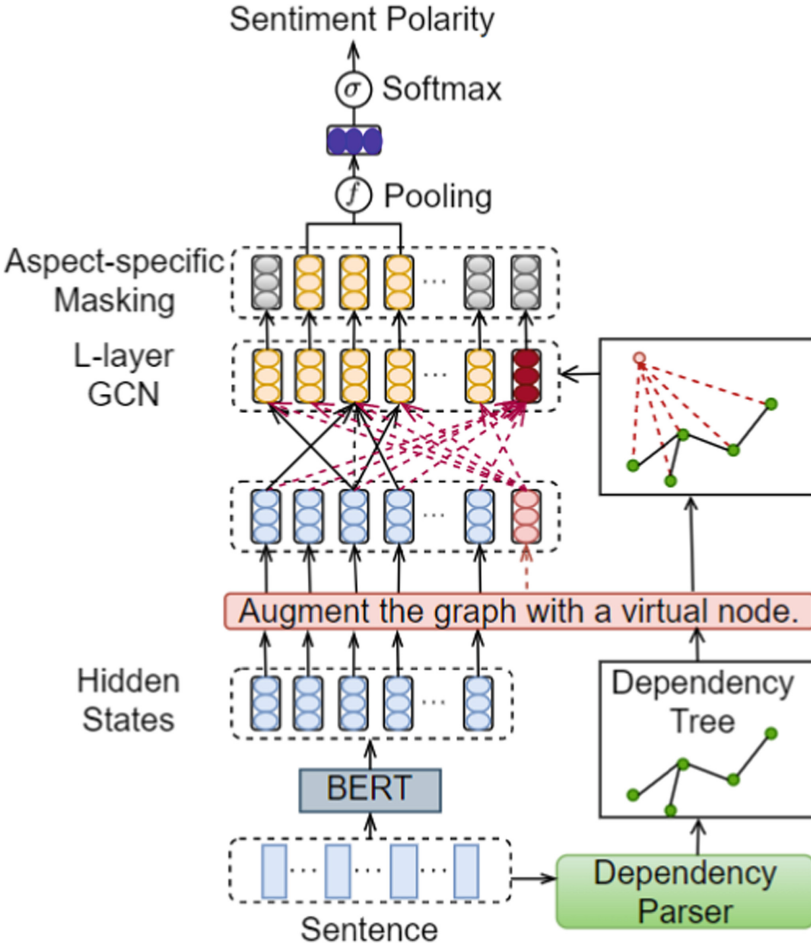
Fig. 2. Overview of the proposed virtual node augmented graph convolutional network.

This paper mainly makes the following contributions: (1) The GCN models applied to the ABSA task are reconsidered so as to exploit the global information regarding the given aspect. (2) A novel ViGCN model for the ABSA task

is proposed in this paper, which can effectively preserve global information via a virtual node. Specifically, the proposed ViGCN model leverages affective commensense knowledge and semantic-relative distance to augment the preserved global information. (3) Extensive experiments on the SemEval 2014 and Twitter datasets demonstrate the superiority of the proposed ViGCN model in ABSA. The code of ViGCN is available at: https://github.com/code4scie/ViGCN.

2   Related Work

Aspect-based sentiment analysis is a fine-grained subtask for sentiment analysis. Early works in this field are feature engineering-based models like SVM [10], which are time- and labor-intensive. Later, deep neural networks have been widely used because of their ability to capture features automatically from the sentences. In general, representative works are based on recursive neural network (RNN) [7,16], long short-term memory (LSTM) [21], convolutional neural network (CNN) [8] and deep memory network [5]. However, these neural networks are generally lacking in a mechanism to leverage the syntax information that is of crucial importance for ABSA [25].

Most of the current state-of-the-art methods are graph network-based models combined with syntactic dependency trees [4,11,19,20,22,25]. For instance, Zhang et al. [25] applied GCNs over dependency trees to exploit word dependencies. Wang et al. [20] proposed an R-GAT model based on an aspect-oriented dependency tree. Chen et al. [4] developed a dotGCN model based on a discrete latent opinion tree. Tang et al. [19] adopted a bidirectional GCN to consider BERT outputs and graph-based representations jointly. Xiao et al. [22] utilized grammatical sequential features from BERT and the syntactic knowledge from dependency graphs to augment GCNs. Li et al. [11] put forward a DualGCN model that contains two GCNs to preserve syntax structure information and semantic correlation message, respectively. Despite exhibiting appealing power in ABSA, the graph network-based models learn node representations mainly based on aggregating features from the neighborhood of each node [14], overlooking the preservation of the global information of the graph that is capable of representing aspect-specific sentiment expression of the sentence. Therefore, how to enhance the node representations of the graph network-based models via effectively leveraging global information should be considered in the task of ABSA.

3   Proposed Model

The overall architecture of the proposed virtual node augmented graph convolutional network (ViGCN) is plotted in Fig. 2. In this section, the details of the ViGCN are presented. In particular, the aspect-based sentiment analysis task is first defined, and then how to initialize the node embeddings of a sentence is illustrated. After that, an introduction is made on how to build a virtual node augmented graph and feed the initial node embeddings and graph to ViGCN. Finally, how to obtain the sentiment polarity and train the model is detailed.

3.1   Task Definition

Given an n-word sentence $S = \{s_1, ..., s_{a_1}, ..., s_{a_k}, ..., s_n\}$ with a k-word aspect $A = \{s_{a_1}, ..., s_{a_k}\}$ included, ABSA is aimed at predicting the sentiment polarity p of the aspect A, where $p \in \{Negative, Neutral, Positive\}$. It is worth noting that the aspect may contain a or several words and $1 \leq k < n$.

3.2   Node Embeddings Initialization

Bidirectional encoder representations from transformers (BERT) [6] is utilized as the aspect-based encoder to learn the hidden contextual representations of sentences. To be specific, under the inspiration of the model LCF-BERT [24], the sentence-aspect pair $G = [CLS] + S + [SEP] + A + [SEP]$ is first constructed as input, so that BERT can capture the semantic relationship between the sentence S and the aspect A through its next-sentence-prediction mechanism [6]. $[CLS]$ and $[SEP]$ are the special tokens of BERT. BERT first tokenizes each word in G into subwords, and the sentence sequence S is tokenized into an m-subword sequence $S_t = \{t_1, ..., t_{a_1}, ..., t_{a_j}, ..., t_m\}$. Then BERT transforms each subword in G into a hidden state vector, and S_t becomes $S_h = \{h_1, ..., h_{a_1}, ..., h_{a_j}, ..., h_m\}$, where $h_i \in \mathbb{R}^{d_h}$ is the hidden state vector of the i-th subword t_i. We use S_h as the initial real node set. Besides, the virtual node is initialized with a zero vector $0 \in \mathbb{R}^{d_h}$. The initial node embeddings are $V^0 = \{h_1, ..., h_{a_1}, ..., h_{a_j}, ..., h_m, 0\}$.

3.3   Construction of the Virtual Node Augmented Graph

For each sentence, a virtual node augmented graph $G = (V, E)$ is constructed to represent the syntactic relationship among subwords, and $A \in \mathbb{R}^{(m+1)\times(m+1)}$ is the adjacency matrix of the graph G. $V = \{x_1, ..., x_{a_1}, ..., x_{a_j}, ..., x_m, x_{m+1}\}$ is the set of nodes. The first part $\{x_1, ..., x_{a_1}, ..., x_{a_j}, ..., x_m\}$ represents the real nodes, and the latter part x_{m+1} indicates the added virtual node. E is the set of edges.

To build the edges between real nodes, a syntactic dependency tree is first constructed for each input sentence using the dependency parsing model LAL-Parser [15]. To match the subword sequence generated by BERT, the syntactic dependency of a word is expanded into all its subwords. Then, an edge is established between two subwords if a dependency is contained in them. Specifically, for $i, j \in [1, m]$:

$$A_{i,j} = \begin{cases} p_e & \text{if } t_i, t_j \text{ may contain dependency,} \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

As suggested by [11], $p_e \in (0, 1]$ here is the probability of a dependency between two subwords from a dependency parser. Its purpose is to reduce the adverse impact of parsing errors on model performance.

Inspired by [23], a virtual node is then added to the graph to preserve global information. Firstly, "naive connections" are constructed between the virtual

node and real nodes. To be specific, following previous work [23], an undirectional edge is established between the virtual node and each real node in the graph. Therefore, for $i \in [1, m]$:

$$A_{i,m+1} = A_{m+1,i} = 1. \tag{2}$$

Like previous work [23], a self-loop is not added to the virtual node, thereby leading to $A_{m+1,m+1} = 0$.

To learn more global information from the words with stronger sentiment, the representation of the adjacency matrix is enhanced by utilizing affective commonsense knowledge from SenticNet. In particular, we use SenticNet 6, which is a public commonsense knowledge base that provides a set of polarity scores associated with 200,000 concepts [2]. Therefore, for $i \in [1, m]$:

$$A_{i,m+1} = A_{m+1,i} = |Sentics(t_i)| + 1, \tag{3}$$

where $Sentics(t_i) \in [-1, 1]$ represents the polarity score of the subword t_i. The value of the polarity score floats between -1 and $+1$. The polarity score of a word is closer to $+1$ when it is more positive. Conversely, the polarity score of a word is closer to -1 when it is more negative. The polarity score of each word in SenticNet 6 is obtained first and then used as the polarity score of its subwords. For those words excluded from SenticNet 6, their polarity scores are set to be 0. Consideration is only given to the strengthening of the connections between the virtual node and real nodes generated from words with an intense sentiment, regardless of whether the sentiment is positive or negative. Therefore, the absolute polarity value is taken.

In LCF-BERT [24], the semantic-relative distance (SRD) is proposed to focus on the subword tokens generated by the local context, given that the local context close to the aspect usually contains significant sentiment information. Under the inspiration of the work of [24], the connections between the virtual node and nodes representing the local context are further enhanced based on SRD:

$$A_{i,m+1} = A_{m+1,i} = \begin{cases} |Sentics(t_i)| + 2 & \text{if } SRD_i < \varphi, \\ |Sentics(t_i)| + 1 & \text{otherwise .} \end{cases} \tag{4}$$

Here, $i \in [1, m]$ and $SRD_i = |P_i - P_a| - \lfloor \frac{k}{2} \rfloor$ represents the SRD between the i-th token and targeted aspect; P_a is the central position of the aspect and P_i is the position of the context word generating the i-th token, respectively; k refers to the length of the aspect sequence; φ stands for the SRD threshold.

3.4   Virtual Node Augmented Graph Convolutional Network

GCN is a special convolutional neural network working directly on graphs and taking advantage of the graph-structured information. After the initialization of node embeddings and the construction of the graph, they are fed into an L-$layer$ GCN to learn local and global information for the given aspect. In an L-$layer$ GCN, $V^l = \left\{ x_1^l, ..., x_{a_1}^l, ..., x_{a_j}^l, ..., x_m^l, x_{m+1}^l \right\}$ ($l \in [1, 2, ..., L]$) is denoted as the

node representations of the *l-th* layer, where $\left\{ x_1^l, ..., x_{a_1}^l, ..., x_{a_j}^l, ..., x_m^l \right\}$ represents real nodes, and x_{m+1}^l is the virtual node. The output of the *i-th* node in the *l-th* layer can be calculated as follows:

$$x_i^l = \sigma \left(\left(\sum_{j=1}^{m+1} A_{ij} W^l x_j^{l-1} \right) / (d_i + 1) + b^l \right), \tag{5}$$

where W^l is a trainable weight matrix and b^l is a bias vector; d_i is the outdegree of the *i-th* node; σ refers to an activation function and *ReLU* is used here. The initial node set is V^0 obtained in Subsect. 3.2.

3.5   Model Training

The final output of the L-layer GCN is $V^L = \left\{ x_1^L, ..., x_{a_1}^L, ..., x_{a_j}^L, ..., x_m^L, x_{m+1}^L \right\}$. Then, the final feature r is obtained through applying average pooling $f_a (\cdot)$ over aspect nodes:

$$r = f_a \left(x_{a_1}^L, ..., x_{a_j}^L \right). \tag{6}$$

Next, the final feature r is fed into a fully connected layer, followed by a softmax layer to learn a sentiment polarity probability distribution p:

$$p = softmax(W_c r + b_c), \tag{7}$$

where W_c and b_c are the trainable weight and bias, respectively.

The model is trained to minimize the objective function composed of a cross-entropy loss function ℓ and an L_2 regularization:

$$\mathfrak{L} = \ell + \lambda \left| \left| \Theta \right| \right|, \tag{8}$$

where Θ represents all the trainable parameters, and λ is the coefficient of L_2 regularization. In this paper, ℓ is defined as follows:

$$\ell = - \sum_{i=1}^{D} \sum_{j=1}^{C} \hat{p}_i^j \log p_i^j, \tag{9}$$

where D is the number of training samples, and C is the number of different sentiment polarities. $\hat{p}$ is the ground-truth sentiment polarity distribution.

4   Experiments

4.1   Datasets and Experiment Settings

We evaluate ViGCN on three public datasets. Restaurant and Laptop datasets are from the SemEval-2014 Task4 [17]. Following [5], all the data samples with the "conflict" label are removed. The Twitter dataset is provided by [7]. All

three datasets contain three sentiment polarities: positive, neural and negative. Table 1 shows the statistics of the datasets.

In this experiment, we use pre-trained BERT-base-uncased model [6] to initialize word embeddings. The dimension of word embeddings and hidden states d_h is 768. The depth of ViGCN layers L is set to be 2, and the dropout rate of ViGCN is set to be 0.1 to avoid overfitting. The SRD threshold φ is set to be 3 on Restaurant and Twitter datasets, and 6 on the Laptop dataset. The Adam optimizer with a learning rate of 0.001 is utilized to optimize the model attributes. The coefficient λ of the L_2 regularization is 10^{-4}. The model is trained in 15 epochs and the batch size is set to be 16. The experimental results are obtained by averaging 10 runs with random initialization, where accuracy and macro F1 score are the evaluation metrics adopted to evaluate the model performance.

Table 1. Statistics of the experimental datasets.

Dataset	Positive		Neural		Negative	
	Train	Test	Train	Test	Train	Test
Restaurant	2164	727	637	196	807	196
Laptop	976	337	455	167	851	128
Twitter	1507	172	3016	336	1528	169

4.2   Comparison Baselines

We compare ViGCN with the following baselines: (1)**ATAE-LSTM** [21] is an attention-based LSTM model for ABSA; (2)**AEN** [18] is an attentional encoder network based on BERT; (3) **RAM** [5] uses a recurrent attention network on memory to learn the sentence representation; (4) **ASGCN** [25] is an aspect-specific GCN model over the dependency tree; (5) **BERT4GCN** [22] is a GCN augmented with intermediate layers of BERT and positional information between words for ABSA task; (6) **R-GAT+BERT** [20] proposes a relational graph attention network based on an aspect-oriented dependency tree; (7) **DualGCN + BERT** [11] integrates syntactic knowledge and semantic information simultaneously with dual GCNs, namely, SynGCN and SemGCN; (8) **DGEDT+BERT** [19] is a dual-transformer model which jointly considers graph-based representations and flat-representations; (9) **dotGCN+BERT** [4] is a graph convolutional network based on a discrete opinion tree.

4.3   Comparison Results

The main experimental results can be seen in Table 2. On all the three experimental datasets, the ViGCN model outperforms almost all compared attention-based and graph neural network-based models with respect to both accuracy

and macro F1 score. Compared to the remarkable DualGCN+BERT model, it performs quite competitively as well. To be specific, ViGCN outperforms Dual-GCN+BERT on Restaurant and Twitter datasets, although its accuracy score on the Laptop dataset is slightly lower than DualGCN+BERT by 0.31. This proves that the ability to effectively preserve global information enables ViGCN to achieve significant gains in ABSA.

Compared with attention-based models such as ATAE-LSTM, AEN and RAM, ViGCN exploits a syntactic dependency tree to explicitly model the connections between the aspect and the context, which thus can avoid the noise introduced by the attention mechanism. In comparison with previous state-of-the-art graph neural network-based models like ASGCN, BERT4GCN and R-GAT+BERT, the enhancement is mainly contributed by two factors. One is that a virtual node is added to the graph because of being able to enhance node representations through leveraging the global information of the whole graph. The other is that effective weights are set for edges between the virtual node and real nodes based on affective commonsense knowledge and semantic-relative distance, which refines the process of the virtual node aggregating and propagating global information.

Table 2. Comparisons of ViGCN with baselines. Accuracy (ACC.) and macro F1 score (F1) are used for metrics. Best results are in bold and second best results are underlined.

Models	Restaurant		Laptop		Twitter	
	Acc.	F1	Acc.	F1	Acc.	F1
ATAE-LSTM	77.20	–	68.70	–	–	–
AEN	84.29	77.22	76.96	73.67	75.14	74.15
RAM	80.23	70.80	74.49	71.35	69.36	67.30
ASGCN	80.77	72.02	75.55	71.05	72.15	70.40
BERT4GCN	84.75	77.11	77.49	73.01	74.73	73.76
R-GAT+BERT	86.60	<u>81.35</u>	78.21	74.07	76.15	74.88
DualGCN+BERT	<u>87.13</u>	81.16	**81.80**	<u>78.10</u>	77.40	76.02
DGEDT+BERT	86.30	80.00	79.80	75.60	77.90	75.40
dotGCN+BERT	86.16	80.49	81.03	<u>78.10</u>	<u>78.11</u>	<u>77.00</u>
ViGCN	**87.31**	**82.27**	<u>81.49</u>	**78.29**	**78.14**	**77.04**

4.4   Ablation Analysis

To further clarify the impact of each component of constructing virtual node augmented graphs for the proposed ViGCN, ablation studies are conducted. The results are demonstrated in Table 3. First, it can be observed that the

model without a virtual node performs poorly on all datasets (ViGCN w/o N+S+D) compared with the other models in Table 3. This indicates virtual node can improve the performance of GCNs in ABSA. When we only construct "naive connections" between the virtual node and real nodes (ViGCN w/o S+D), the performance of the model, although improved, is still far below the best performance. Comparatively, ViGCN w/o D and ViGCN w/o S evidently perform better. This proves the effectiveness of the SenticNet-based and SRD-based optimization methods for "naive connection". Also, the "naive connections" are indispensable to the model, given that the removal of "naive connections" leads to poorer performance (ViGCN w/o N). It is worth noting that ViGCN w/o D does not outperform ViGCN w/o S on the Twitter dataset as on the Restaurant and Laptop datasets, which may be because the data on the Twitter dataset is biased towards colloquial expressions, less sensitive to sentiment information [7]. Finally, ViGCN outperforms all its ablation models, revealing that each component of ViGCN is indispensable.

Table 3. Results of ablation analysis. "S" represents SenticNet, "D" represents SRD, "N" represents "naive connections".

Models	Restaurant		Laptop		Twitter	
	Acc.	F1	Acc.	F1	Acc.	F1
ViGCN w/o S+D+N	85.08	78.41	78.80	74.31	75.63	74.83
ViGCN w/o D+N	85.43	78.64	79.91	75.56	75.92	74.95
ViGCN w/o S+N	85.61	78.62	80.06	75.95	76.07	75.47
ViGCN w/o S+D	85.52	78.15	80.22	76.28	76.37	75.50
ViGCN w/o N	85.70	79.06	80.54	76.62	76.81	75.40
ViGCN w/o S	86.06	80.18	80.85	77.10	77.40	76.03
ViGCN w/o D	86.95	80.94	81.01	77.73	77.10	75.77
ViGCN	**87.31**	**82.27**	**81.49**	**78.29**	**78.14**	**77.04**

4.5 Parameter Sensitivity

Figure 3 illustrates the performance of ViGCN at different SRD thresholds φ from 1 to 10. It can be seen that ViGCN performs best with an SRD threshold of 3 on the Restaurant and Twitter datasets, and with an SRD threshold of 6 on the laptop dataset. As the SRD threshold φ increases, the performance of the model gradually improves until it reaches the best performance, and then the performance of the model shows a drop trend. A possible reason is that when φ is too small, the model cannot capture enough information from the local context. On the opposite, while it is too large, noise may be introduced into the global information preserved by the model.
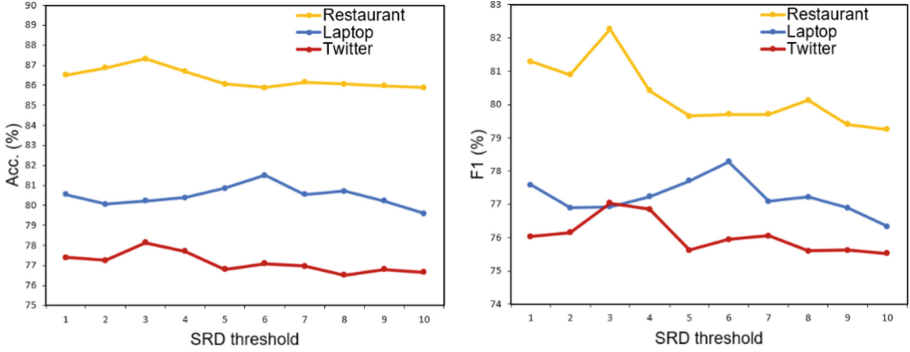
Fig. 3. Effect of the SRD threshold φ.

4.6   Case Study

ViGCN, RAM and ASGCN are compared on several sample cases. The results are demonstrated in Table 4. For the first sample, the words "apple" and "OS" in the aspect"apple OS" are five and four hops away, respectively, from the opinion word "happy" on the graph over the dependency tree. For a 2-layer ASGCN, the information contains in the opinion word cannot be passed to the aspect. In contrast, in ViGCN, information can be passed from the opinion word to the aspect in 2 hops via the virtual node. Given this, ViGCN succeeds while ASGCN fails. For the second sample, the attention-based model RAM wrongly predicts the sentiment polarity of the aspect "Saketini", which may attend to the noise word "Disappointingly". For the third sample, impacted by the noise word "wonderful", ASGCN and RAM mispredict the sentiment polarity of the aspect "burger". Nonetheless, ViGCN predicts it correctly because the model combines global information preserved from the entire sentence to make predictions, thereby significantly mitigating the interference of those noise words.

Table 4. Case study. The words in red denote the aspects. The symbols P, N and O represent positive, negative and neural sentiment, respectively.

Sentence	ViGCN	RAM	ASGCN
1. Works well, and I am extremely happy to be back to an apple OS.	P✓, P✓	P✓, P✓	P✓, O✗
2. Disappointingly, their wonderful Saketinihas been taken off the bar menu.	P✓	N✗	P✓
3. My friend had a burger and I had these wonderful blueberry pancakes.	O✓, P✓	P✗, P✓	P✗, P✓

5   Conclusion

In this paper, the task of aspect-based sentiment analysis is investigated and a virtual node augmented graph convolutional network called ViGCN is proposed.

Taking advantage of the virtual node, the proposed ViGCN can effectively preserve global information to precisely predict the sentiment polarity towards a given aspect. Empirical results on three public datasets demonstrate the effectiveness of our model. Future work includes applying the virtual node to other graph-based models in ABSA, e.g., graph attention network [20].

References

1. Buffelli, D., Vandin, F.: The impact of global structural information in graph neural networks applications. Data p. 10 (2022)
2. Cambria, E., Li, Y., Xing, F.Z., Poria, S., Kwok, K.: Senticnet 6: Ensemble application of symbolic and subsymbolic ai for sentiment analysis. In: Proceedings of the 29th ACM International Conference on Information and Knowledge Management, pp. 105–114 (2020)
3. Cao, S., Lu, W., Xu, Q.: Grarep: Learning graph representations with global structural information. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 891–900 (2015)
4. Chen, C., Teng, Z., Wang, Z., Zhang, Y.: Discrete opinion tree induction for aspect-based sentiment analysis. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, pp. 2051–2064 (2022)
5. Chen, P., Sun, Z., Bing, L., Yang, W.: Recurrent attention network on memory for aspect sentiment analysis. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 452–461 (2017)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4171–4186 (2019)
7. Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., Xu, K.: 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 (2014)
8. Fan, C., Gao, Q., Du, J., Gui, L., Xu, R., Wong, K.F.: Convolution-based memory network for aspect-based sentiment analysis. In: The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1161–1164 (2018)
9. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry. In: International Conference on Machine Learning, pp. 1263–1272 (2017)
10. Jiang, L., Yu, M., Zhou, M., Liu, X., Zhao, T.: Target-dependent twitter sentiment classification. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 151–160 (2011)
11. Li, R., Chen, H., Feng, F., Ma, Z., Wang, X., Hovy, E.: Dual graph convolutional networks for aspect-based sentiment analysis. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pp. 6319–6329 (2021)
12. Liu, M., Liu, Y.: Inductive representation learning in temporal networks via mining neighborhood and community influences. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2202–2206 (2021)

13. Liu, M., Quan, Z.W., Wu, J.M., Liu, Y., Han, M.: Embedding temporal networks inductively via mining neighborhood and community influences. Appl. Intell. 1–20 (2022)

14. Liu, M., Wu, J.M., Liu, Y.: Embedding global and local influences for dynamic graphs. In: CIKM (2022)

15. Mrini, K., Dernoncourt, F., Tran, Q.H., Bui, T., Chang, W., Nakashole, N.: Rethinking self-attention: Towards interpretability in neural parsing. In: Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 731–742 (2020)

16. Nguyen, T.H., Shirai, K.: Phrasernn: Phrase recursive neural network for aspect-based sentiment analysis. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp, 2509–2514 (2015)

17. Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., Manandhar, S.: Semeval-2014 task 4: Aspect based sentiment analysis. In: COLING 2014 (2014)

18. Song, Y., Wang, J., Jiang, T., Liu, Z., Rao, Y.: Attentional encoder network for targeted sentiment classification. arXiv preprint arXiv:1902.09314 (2019)

19. Tang, H., Ji, D., Li, C., Zhou, Q.: 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 (2020)

20. Wang, K., Shen, W., Yang, Y., Quan, X., Wang, R.: 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 (2020)

21. Wang, Y., Huang, M., Zhu, X., Zhao, L.: Attention-based lstm for aspect-level sentiment classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 606–615 (2016)

22. Xiao, Z., Wu, J., Chen, Q., Deng, C.: BERT4GCN: Using BERT intermediate layers to augment GCN for aspect-based sentiment classification. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 9193–9200 (2021)

23. Ying, C., et al.: Do transformers really perform badly for graph representation? In: Advances in Neural Information Processing Systems, vol. 34 (2021)

24. Zeng, B., Yang, H., Xu, R., Zhou, W., Han, X.: Lcf: A local context focus mechanism for aspect-based sentiment classification. Appl. Sci. **9**(16), 3389 (2019)

25. Zhang, C., Li, Q., Song, D.: 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, pp. 4568–4578 (2019)