

Dual-Channel Interactive Graph Convolutional Networks for Aspect-Level Sentiment Analysis

Zhouxin Lan ¹, Qing He ^{1,*} and Liu Yang ^{2,*}¹ College of Big Data and Information Engineering, Guizhou University, Guiyang 550025, China² School of Public Administration, Guizhou University, Guiyang 550025, China

* Correspondence: qhe@gzu.edu.cn (Q.H.); lyang3@gzu.edu.cn (L.Y.)

Abstract: Aspect-level sentiment analysis aims to identify the sentiment polarity of one or more aspect terms in a sentence. At present, many researchers have applied dependency trees and graph neural networks (GNNs) to aspect-level sentiment analysis and achieved promising results. However, when a sentence contains multiple aspects, most methods model each aspect independently, ignoring the issue of sentiment connection between aspects. To address this problem, this paper proposes a dual-channel interactive graph convolutional network (DC-GCN) model for aspect-level sentiment analysis. The model considers both syntactic structure information and multi-aspect sentiment dependencies in sentences and employs graph convolutional networks (GCN) to learn its node information representation. Particularly, to better capture the representations of aspect and opinion words, we exploit the attention mechanism to interactively learn the syntactic information features and multi-aspect sentiment dependency features produced by the GCN. In addition, we construct the word embedding layer by the *BERT* pre-training model to better learn the contextual semantic information of sentences. The experimental results on the restaurant, laptop, and twitter datasets show that, compared with the state-of-the-art model, the accuracy is up to 1.86%, 2.50, 1.36%, and 0.38 and the *Macro-F1* values are up to 1.93%, 0.61%, and 0.4%, respectively.

Citation: Lan, Z.; He, Q.; Yang, L. Dual-Channel Interactive Graph Convolutional Networks for Aspect-Level Sentiment Analysis. *Mathematics* **2022**, *10*, 3317. <https://doi.org/10.3390/math10183317>

Academic Editor: Pedro A. Castillo Valdivieso

Received: 16 August 2022

Accepted: 8 September 2022

Published: 13 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: aspect-level sentiment analysis; graph neural network; attention mechanism; *BERT*

MSC: 68T50; 03B65

1. Introduction

With the rapid development of social networks and internet applications, an increasing amount of user comment data are emerging on the web. Usually, the emotional tendencies of these data are of great value in understanding users and businesses and mining user emotions. However, due to the complex structure of online review data, the performance of traditional sentiment analysis cannot achieve our expected results. Therefore, researchers propose a new task named aspect-based sentiment analysis [1,2]. Different from traditional coarse-grained sentiment analysis, aspect-based sentiment analysis is an entity-oriented, fine-grained sentiment analysis task that aims to identify the sentiment polarity of specific aspects in sentences (e.g., negative, neutral, and positive). Taking “*the restaurant food is good but the service is dreadful*” as an example, the emotional polarities of the aspect words “*restaurant food*” and “*service*” in the sentence are positive and negative, respectively. Therefore, aspect-based sentiment analysis can accurately identify the user’s attitude toward an aspect, rather than simply judging the sentiment polarity of a sentence. Aspect-based sentiment analysis mainly includes two stages: aspect-level word extraction [3,4] and aspect-level sentiment classification [5]. The former extracts the questions in the comments, and the latter classifies the opinions of these questions. In this paper, we mainly focus on sentiment classification.

The key to aspect-level sentiment analysis tasks is to model the connections between aspect words and their corresponding opinion words. Early sentiment analysis methods mainly used handcrafted features combined with traditional machine learning methods [6–8]. However, these methods heavily depend on the quality of handcrafted features in performance and have reached a bottleneck stage. In recent years, deep learning methods represented by neural networks have attracted increasing attention because they can automatically generate useful feature representations from aspects and their contexts, and can achieve better aspect-level sentiment classification without handcrafted features. In particular, the attention mechanism [9,10] and graph neural network [11–15] methods are widely used in aspect-level sentiment classification due to their ability to focus on aspect words in sentences and their ability to handle unstructured data [16–23]. For example, Su et al. [16] proposed a progressive self-supervision attention learning approach for attentional aspect-level sentiment analysis. Wu et al. [17] adopted a multi-head attention mechanism to generate aspect and context feature representations. Zhang et al. [22] used graph convolutional networks to learn node information in dependency trees and combined attention mechanisms for sentiment classification tasks. Xiao et al. [23] constructed a syntactic edge-enhanced graph convolutional network by considering the different types of neighborhoods with the edge constraint.

In this paper, we propose a GCN based method combining syntactic and multi-aspect sentiment features. We construct a multi-aspect sentiment dependency graph and elaborately design an information interaction layer to enhance the ability to precisely capture the semantic associations of different aspects with context.

The main contributions from this research are as follows.

- We propose a novel aspect-level sentiment classification approach exploiting both syntactic dependency trees and multi-aspect sentiment graphs to better capture the representation between aspects and opinion words. In addition, to better learn the context semantic information of sentences, we also introduce the *BERT* pre-training model to build the word embedding layer.
- We propose a novel architecture, which consists of two kinds of GCNs, to effectively encode both syntactic dependency trees and multi-aspect sentiment graphs. Moreover, the two kinds of information are learned interactively using the attention mechanism to obtain more informative representations.
- We evaluate DC-GCN on three datasets, namely twitter, laptop, and restaurant. Experiments show that the proposed model achieves superior performance over the state-of-the-art approaches.

The remainder of the paper is organized as follows: Section 2 introduces related work. Section 3 describes the details of the GCN. Section 4 describes the details of the proposed DC-GCN model. Section 5 analyses the experimental results of the proposed DC-GCN model. The conclusion of this work is provided in Section 6.

2. Related Work

Aspect-level sentiment analysis, also known as opinion mining, is a fine-grained task in natural language processing (NLP). Early work mainly adopted traditional machine learning aspects for sentiment classification. However, machine learning methods with supervised classifiers heavily rely on the quality of large handcrafted features [24]. In recent years, with the development in the field of neural network natural language processing, an increasing number of neural-network-based methods have been proposed to solve sentiment analysis tasks, such as recurrent neural networks (RNN) [25], convolutional neural networks (CNN) [26] and memory neural networks (MNN) [27]. Dong et al. [28] proposed an adaptive recurrent neural network to adaptively learn the tree structure of sentences. Xue et al. [29] proposed using convolutional neural networks to obtain feature representations of aspects and contexts, and to obtain the relationship between contexts and aspects through a gating mechanism. Tang et al. [30] proposed a

memory-based neural network model to store important information in specific aspects. In addition, long short-term memory networks (LSTMs) have been widely used in aspect-level sentiment analysis due to their ability to effectively capture long-term dependencies and solve the vanishing gradient problem. Tang et al. [31] proposed a target-dependent long short-term memory (TD-LSTM) model. This method uses two LSTMs to model the context on the left and right sides of an aspect, and the last hidden state of the spliced two LSTMs is used as the classification feature of aspect-level sentiment analysis.

Subsequently, researchers found that the attention mechanism can effectively identify those words in a sentence that are more important in a certain aspect and are widely used in this task. Wang et al. [32] proposed an attention-based ATAE-LSTM model. This method introduces attention into LSTMs for the first time, and the experimental results show that the model with added attention is significantly better than LSTMs. Ma et al. [33] found that the same aspect expresses different emotional tendencies in different contexts and proposed an interactive attention network (IAN) model. This method further improves the accuracy of aspect-level sentiment analysis tasks by interactively learning context and aspect representations. Huang et al. [34] introduced an attention-over-attention mechanism based on IAN and proposed an AOA-LSTM model without pooling layers. The model can effectively capture the interaction information between aspects and contexts in sentences. Although the neural network model that introduces the attention mechanism can effectively improve the classification effect of aspect-level sentiment analysis, when the aspect consists of multiple words, the attention-based model may mistakenly focus on the context words that are not related to the aspect.

Recently, graph neural networks (GNNs) have been widely used to mine the syntactic structure information of sentences due to their excellent performance and high interpretability. Zhang et al. [22] used graph convolutional neural networks (GCNs) for the first time to obtain syntactic dependency information between contexts and aspects in syntactic dependency trees. Zhao et al. [35] proposed a GCN-based aspect-level sentiment analysis model, which can effectively capture the emotional dependencies of multiple aspects in sentences. Zhu et al. [36] proposed a global- and local-dependency-guided graph convolutional network (GL-GCN) model by combining global and local structural information. Miao et al. [37] proposed a contextual graph attention network (CGAT) model to address the problems of insufficient semantic information extraction and high computational complexity of the attention mechanism.

Although the above methods have achieved certain success, most of these methods model each aspect independently, which makes the hidden contact information of multiple aspects in a sentence underutilized. For example, *"The location of this restaurant is convenient and the food is delicious, but the serving speed is too slow"*. The sentiment polarity of the first aspect "location" is positive. From the conjunction "and", we easily know that the second aspect "food" and "location" have the same polarity. Similarly, from the connection "but", we can infer that the sentiment polarity of the last aspect is negative. Meanwhile, the syntactic dependency tree contains rich syntactic relation and distance information. However, most of the current graph-neural-network-based models are modelled around syntactic dependency trees, and there is no model that considers both syntactic dependency trees and multi-aspect sentiment dependencies. Therefore, to address the deficiencies of existing models, we propose a dual-channel interactive graph convolutional network for the aspect-based sentiment analysis (DC-GCN) model, which combines a syntactic dependency tree and a multi-aspect sentiment dependency graph.

3. Graph Convolutional Network (GCN)

3.1. Syntactic Dependency Tree

We employ the dependency parser in the Spacy toolkit to generate syntactic dependency trees for sentences. Taking “The food is okey while the atmosphere is not really good” as an example, its dependency tree structure is shown in Figure 1. As seen from the figure, green fonts represent aspect words, red fonts represent sentiment words, directed edges represent dependencies, and labels represent dependency types. In this paper, to fully exploit the syntactic information of syntactic structure, we use a graph convolutional neural network to encode a syntactic dependency tree to capture the syntactic dependencies of aspects and contexts.

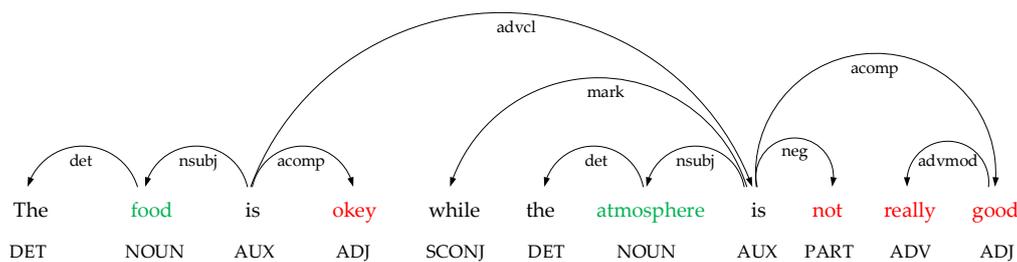


Figure 1. Syntactic dependency tree.

3.2. Graph Convolutional Network

The graph convolutional network (GCN) is a variant of the GNN, which is inspired by the traditional convolutional neural network (CNN) and graph embedding to encode local features of unstructured data [38]. An example of a single-layer GCN is shown in Figure 2. Given a dependency graph with n nodes, the graph can be represented as an adjacency matrix $A \in R^{n \times n}$. For convenience, we set the feature vector of any node i in the l -layer GCN to be $h_i^l, l \in [1, 2, \dots, L]$. Therefore, the mathematical model of graph convolution operating on node representation is described as follows.

$$h_i^l = \sigma(\sum_{j=1}^N A_{ij} h_j^{l-1} W^l + b^l) \tag{1}$$

where W^l is a weight matrix, b^l is a bias term, and σ is an activation function (e.g., ReLU).

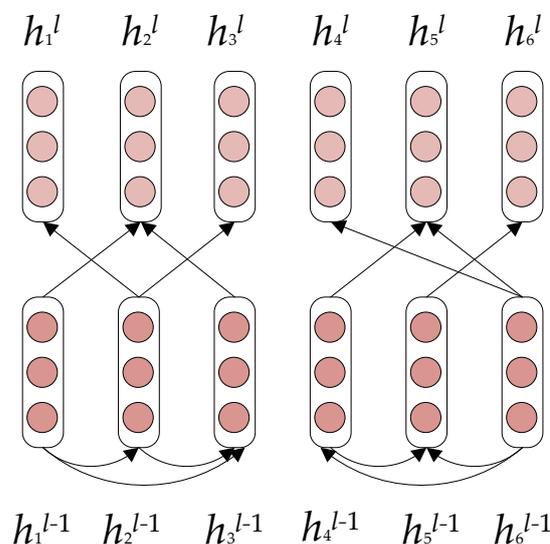


Figure 2. An example of a single-layer GCN.

4. Methodology

Given a sentence $S = \{w_1, \dots, w_t, \dots, w_{t+m}, \dots, w_n\}$ of length n and an aspect $a = \{w_t, \dots, w_{t+m}\}$ of length m , where the aspect word a is a subsequence of sentence S , the overall structure of the proposed DC-GCN model is shown in Figure 3. The DCGCN model consists of five parts, namely the word embedding layer, the position encoding layer, the graph convolutional network layer, the information interaction layer and the output layer. The individual components of the model are described separately in the remainder of this section.

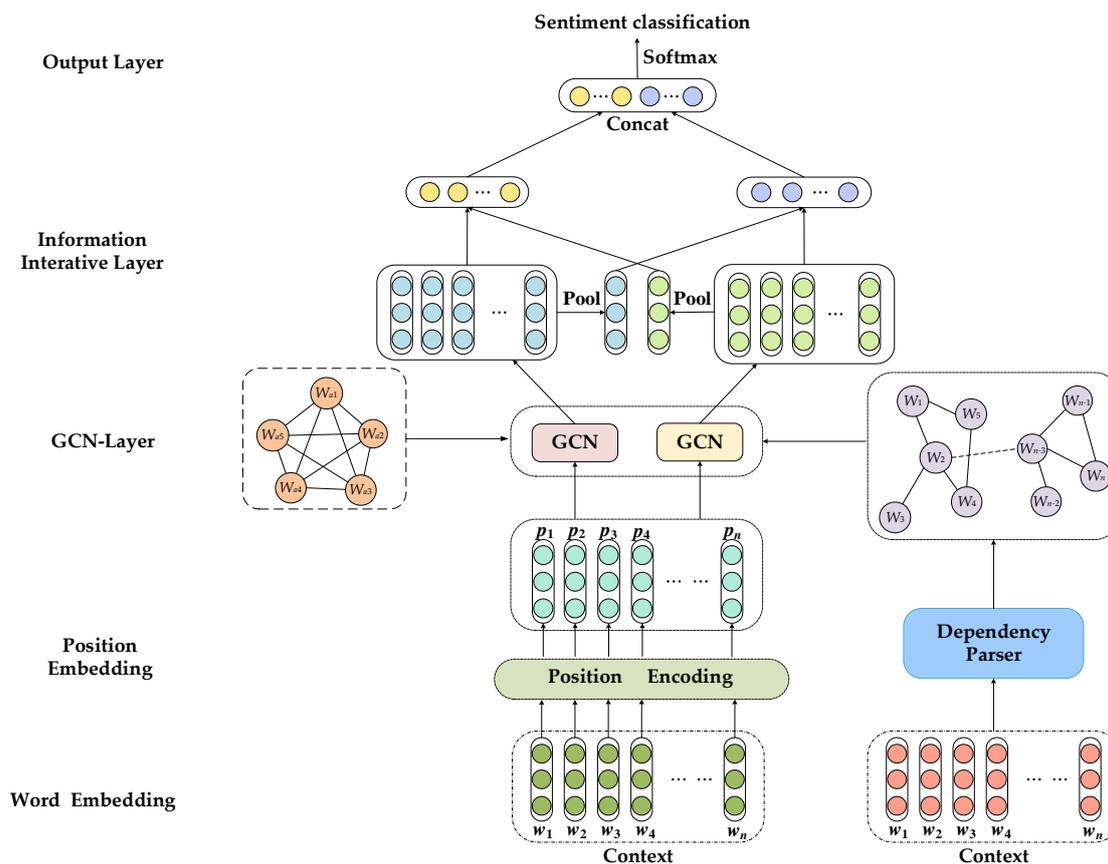


Figure 3. The overall framework of the proposed DC-GCN model.

4.1. Word Embedding Layer

Word embeddings map each word w_i in a given sentence to a low-dimensional real-valued vector space. This paper employs the pre-trained language model *BERT* [39] to vectorize the input sentence. The *BERT* pre-training model is a bidirectional Transformer model that can infer semantic information according to the contextual relationship of words, effectively solving the problem of polysemy. There are two variants of the *BERT* model, named *BERT_{base}* and *BERT_{large}*. The model used in this paper is *BERT_{base}*, and its structure is shown in Figure 4.

To facilitate the training and fine-tuning of the *BERT* model, we define the context and aspect words as “[CLS] + sentence + [SEP] + aspect + [SEP]” as the input of the model and take the output of the last layer of *BERT* as the next layer input. The mathematical model can be described as follows:

$$h_i = BERT(w_i) \quad i = 1, 2, \dots, n \tag{2}$$

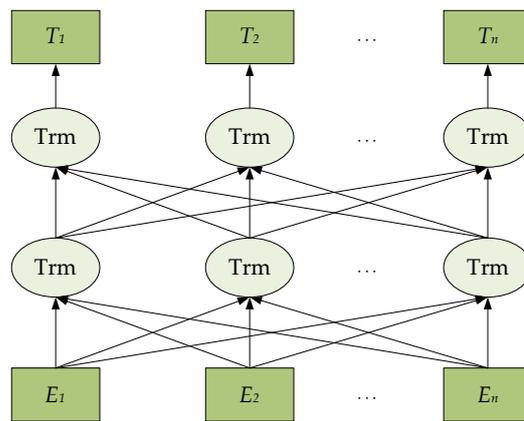


Figure 4. Structure of BERT.

4.2. Position Embedding Layer

The polarity of the target aspect is susceptible to context words that are closer to the aspect. Therefore, we introduce positional encoding to enhance the informative features of context words that are closer to the aspect and weaken the information of context words that are farther away from the aspect. The position encoding information can be obtained from the path length of aspect and context words in the sentence, and its mathematical model is defined as follows.

$$p_i = \begin{cases} 1 - \frac{a_{\text{start}} - i}{n} & 0 \leq i < a_{\text{start}} \\ 0 & a_{\text{start}} \leq i < a_{\text{end}} \\ 1 - \frac{i - a_{\text{end}}}{n} & a_{\text{end}} \leq i \leq n \end{cases} \quad (3)$$

where a_{start} and a_{end} represent the start index and end index of the aspect, respectively; n is the length of the sentence; and p_i represents the relative distance from the context word w_i in the sentence to the aspect. Finally, we can obtain the sentence feature vector with location information.

$$h_i = h_i \cdot p_i \quad (4)$$

4.3. DC-GCN Layer

To capture the syntactic information and multi-aspect sentiment dependency information in sentences, we employ a dual-channel graph convolutional layer in the model. This layer consists of two modules: a syntactic-based graph convolutional network and a multi-aspect dependency-based graph convolutional network. The former model is used to obtain the rich syntactic information in the dependency tree, and the latter is used to obtain the multi-aspect sentiment dependency information in the sentence.

4.3.1. Syntactic Graph Convolution Module

In this section, we construct a contextual syntactic graph $G_{\text{sy}}(V, E)$, where node V represents the hidden state of a word and edge E represents the syntactic dependency between words. As mentioned above, we use the Spacy toolkit to obtain the dependency tree of the sentence. The weight between the v_i and v_j is defined as:

$$A_{ij}^{sy} = \begin{cases} 1 & \text{if } e_{ij} \in E \\ 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where A_{ij} denotes an adjacency matrix, $i, j \in [1, n]$, and $e_{ij} \in E$ denotes when node i depends on node j .

Then, we use the graph convolutional network to extract the syntactic information, and the node representation is calculated as follows:

$$h_i^{sy(l)} = \text{RELU} \left(\sum_{j=1}^n \tilde{A}_{ij}^{sy} h_j^{l-1} W^l + b^l \right) \tag{6}$$

where $\tilde{A}^{sy} = \tilde{D}^{-\frac{1}{2}} A^{sy} \tilde{D}^{\frac{1}{2}}$ is the normalized symmetric adjacency matrix, \tilde{D} is the degree matrix of \tilde{A}^{sy} , h_j^{l-1} is the word representation of the position information added to the previous layer, $h_i^{sy(l)}$ is the final representation of the i -th node of the first layer of the GCN, and W^l and b^l are the weight matrix and bias term of the GCN first layer, respectively.

4.3.2. Multi-Aspect Sentiment Graph Convolution Module

As we discuss in Section 1, existing models mostly model each aspect independently, ignoring the hidden connection information among multiple aspects. To address this problem, we construct a multi-aspect sentiment graph to capture dependency information between aspects in one sentence. As shown in Figure 3, the nodes of the multi-aspect sentiment graph are determined only by the number of aspects, where the edges represent the dependencies between aspects.

Then, we employ a graph convolutional network to extract the hidden information representation of the multi-aspect sentiment graph, and the node representation is computed as follows:

$$h_i^{ma(l)} = \text{RELU} \left(\sum_{j=1}^n \tilde{A}_{ij}^{ma} h_j^{l-1} W^l + b^l \right) \tag{7}$$

where $\tilde{A}^{ma} = \tilde{D}^{-\frac{1}{2}} A^{ma} \tilde{D}^{\frac{1}{2}}$ is the normalized symmetric adjacency matrix, \tilde{D} is the degree matrix of \tilde{A}^{ma} , h_j^{l-1} is the word representation of the position information added to the previous layer, $h_i^{ma(l)}$ is the final representation of the i -th node of the first layer of the GCN, and W^l and b^l are the weight matrix and bias term of the GCN first layer, respectively.

4.4. Information Interactive Layer

Considering the interactive information between aspects and contexts is also important for sentiment prediction. In this section, we use an attention mechanism to model the information representation between context and multi-aspect terms in a sentence, generating aspect-oriented and context-oriented representations, respectively. The context vectors with syntactic information are first average pooled, and then the attention weight β_i for each aspect is calculated. The mathematical model is defined as follows.

$$h_{avg}^{sy} = \frac{1}{n} \left(\sum_{i=1}^n h_i^{sy} \right) \tag{8}$$

$$\alpha(h_{avg}^{sy}, h_i^{ma}) = \tanh(h_{avg}^{sy} \cdot W \cdot h_i^{maT} + b) \tag{9}$$

$$\beta_i = \frac{\exp(\alpha(h_{avg}^{sy}, h_i^{ma}))}{\sum_{j=1}^n \exp(\alpha(h_{avg}^{sy}, h_j^{ma}))} \tag{10}$$

where, α is the score function, h_i^{maT} is the transpose of h_i^{ma} , β_i is the assigned weight, and W and b are the weight matrix and bias term, respectively.

For context effects, we first average pool the vectors with multi-aspect dependent features, and then compute the attention weight φ_i for each contextual word. The mathematical model is described as follows.

$$h_{avg}^{ma} = \frac{1}{m} \left(\sum_{i=1}^m h_i^{ma} \right) \tag{11}$$

$$\phi(h_{avg}^{ma}, h_i^{sy}) = \tanh(h_{avg}^{ma} \cdot W \cdot h_i^{syT} + b) \tag{12}$$

$$\varphi_i = \frac{\exp(\alpha(h_{avg}^{ma}, h_i^{sy}))}{\sum_{j=1}^n \exp(\alpha(h_{avg}^{ma}, h_j^{sy}))} \tag{13}$$

Then, the weighted multi-aspect and context representations are calculated by Equations (14) and (15) below.

$$h_a = \sum_{i=1}^m \beta_i \cdot h_i^{ma} \tag{14}$$

$$h_c = \sum_{i=1}^n \varphi_i \cdot h_i^{sy} \tag{15}$$

Finally, the generated multi-aspect feature vector h_a is spliced with the context feature vector h_c to obtain $r = [h_a; h_c]$.

4.5. Output Layer

To predict the sentiment polarity for a given aspect, we input the final representation r obtained by the information interaction layer into a fully connected layer, which is finally output by softmax for classification. The mathematical model of the predicted probability p of sentiment polarity is defined as follows.

$$p = \text{softmax}(W_p r + b_p) \tag{16}$$

where $p \in R^{d_p}$ is the polarity decision space, d_p represents the number of categories of sentiment polarity, and W_p and b_p are the weight parameter and bias term, respectively.

The training of the model adopts the cross-entropy loss function and L_2 -regularization, and its calculation formula is as follows.

$$Loss = -\sum_{i=1}^{d_p} y_i \log(p_i) + \lambda \|\theta\|_2 \tag{17}$$

where d_p represents the number of categories of sentiment polarity, y_i represents the true sentiment polarity, p_i is the predicted sentiment polarity, and λ is the L_2 -regularization coefficient, which is the parameter that the model needs to train.

5. Experiments

5.1. Datasets and Evaluation Metrics

To verify the effectiveness of the DC-GCN model, we conduct experiments on three public datasets, namely Restaurant and Laptop for the SemEval-2014 task [40] and the Twitter dataset curated by Dong et al. [28] Each sample of the dataset is a sentence actually generated by the user. Sentences are marked with sentiment labels of one or more aspects, among which sentiment labels have three categories: positive, negative, and neutral. The statistical results of the dataset are shown in Table 1.

Table 1. Detailed statistical of the three experimental datasets.

Dataset	Positive		Neutral		Negative	
	Train	Test	Train	Test	Train	Test
Restaurant	2164	728	637	196	807	196
Laptop	994	341	464	169	870	128
Twitter	1561	173	3127	346	1560	173

We employ two evaluation metrics, Accuracy (Acc) and Macro-F1 (MF1), to evaluate the effectiveness of the DC-GCN model. For a single category, let TP be the number of correctly predicted samples, FP be the number of samples predicted to be the current category from other categories, and FN be the number of samples predicted to be other categories from the current category. Then, the calculation formulas for accuracy and MF1 are distributed as follows in (18) and (19).

$$\text{Acc} = \frac{\sum_{i=1}^{d_p} TP_i}{\sum_{i=1}^{d_p} (TP_i + FP_i + FN_i)} \quad (18)$$

$$\left\{ \begin{array}{l} \text{MF1} = \frac{1}{d_p} \sum_{i=1}^{d_p} \frac{2 \cdot P_i \cdot R_i}{P_i + R_i} \\ P = \frac{TP}{TP + FN} \\ R = \frac{TP}{TP + FP} \end{array} \right. \quad (19)$$

where d_p represents the number of categories of sentiment polarity, P is the precision rate, and R is the recall rate.

5.2. Experimental Settings

In our experiments, we employ a 768-dimensional pre-trained *BERT* model to generate word embedding vectors. All weight matrices are randomly initialized by a uniform distribution $U(-0.01, 0.01)$, and all biases are set to 0. The batch size is set at 32, and the max length of a sentence is set to 85. The number of layers of the GCN is set to 2. The model uses the Adam optimizer to optimize the model parameters during the training process. The learning rate and L_2 -regularization adjustment coefficients are 2×10^{-5} and 1×10^{-5} , respectively. In addition, to avoid overfitting, the dropout rate is set to 0.3. Hyperparameter settings are shown in Table 2. We mainly use the PyTorch deep learning framework to train and predict the model. All the experiments are conducted on hardware with an Intel Core CPU i5-7500 3.6 GHz and an NVIDIA GTX 3090.

Table 2. Hyper-parameter settings.

Parameters	Values
Word embedding dimensions	768
GCN layers	32
Dropout rate	0.3
Learning rate	2×10^{-5}
L_2 -regularization coefficients	1×10^{-5}

5.3. Baseline Models

We compare DC-GCN with state-of-the-art baselines. The models are briefly described as follows.

ATAE-LSTM [32] concatenates the aspect word feature vector and context feature vector as input and then uses an attention-added LSTM to capture the connection between aspect and context.

IAN [33] uses two LTSMs and an attention mechanism to interactively learn aspect and context representations.

RAM [21] uses an attention-based aggregation model to learn sentence representations with multi-layered structures.

AEN [41] uses an attention-based encoder to model the relationship between a given aspect and its corresponding context to avoid duplication.

ASGCN [22] uses a GCN for the first time to obtain node information on the syntactic dependency tree and combines the attention mechanism for sentiment classification.

ASEGCN [23] utilizes bidirectional LSTM and the multi-head attention mechanism to generate semantic codes, and introduces a GCN into the model.

GL-GCN [36] employs two GCN models to independently encode global and local structural information, and adaptively fuses the two types of information using a gating mechanism.

SK-GCN1 + BERT [42] uses two GCNs to independently model dependency trees and knowledge graphs, combined with multi-head attention for aspect-level sentiment classification.

SK-GCN2 + BERT [42] uses a GCN to jointly model syntax and knowledge graphs, combined with multi-head attention for aspect-level sentiment classification.

R-GAT + BERT [43] reconstructs an aspect-oriented syntactic dependency tree and utilizes a graph attention network (GAT) to learn feature representations.

CGAT + BERT [37] employs two GATs to aggregate syntactic structure information into target aspects and combines contextual attention networks to extract semantic information in sentence-aspect sequences.

5.4. Results and Analysis

In this section, we compare the aspect-level sentiment analysis tasks of the DC-GCN and other models on three datasets, where the experimental results of each comparison model are derived from their original papers. The experimental results are shown in Table 3, in which the bold font represents the best results in the current indicators and “-” represents missing data in the literature.

It can be observed from the data in Table 3 that our proposed DC-GCN model consistently outperforms other models in the literature on the Restaurant, Laptop, and Twitter datasets, which proves the superiority of our model. Specifically, the DC-GCN model improves the accuracy by up to 1.86% and the MF1 value by 1.93% on the Restaurant dataset. In addition, the DC-GCN model improves the accuracy by up to 1.36% and the MF1 value by 0.61% on the Laptop dataset. However, the performance improvement is the least on the Twitter dataset, with only 0.38% and 0.40% improvement in accuracy and MF1 value, respectively. Because the syntactic information of the Twitter dataset is more chaotic than the other two kinds of data, it makes the modules in the DC-

GCN model that rely on syntactic information unable to effectively capture its features. In summary, our proposed DC-GCN model outperforms all baselines. The experimental results demonstrate that the fusion of grammatical structure information and multi-aspect sentiment dependency information is crucial. DC-GCN can effectively capture two kinds of structural information through two kinds of GCNs and fuse the two kinds of structural information through the information interaction layer to mutually compensate for the lack of information of a single structure.

Table 3. Comparison results of different models on the three datasets.

Model	Restaurant		Laptop		Twitter	
	Acc (%)	MF1 (%)	Acc (%)	MF1 (%)	Acc (%)	MF1 (%)
ATAE-LSTM [32]	77.20	-	68.90	-	-	-
IAN [33]	78.60	-	72.10	-	-	-
RAM [21]	80.23	70.80	74.49	71.35	69.36	67.30
AEN [41]	80.98	72.14	73.51	69.04	72.83	69.81
ASGCN [22]	80.77	72.02	75.55	71.05	72.15	70.40
ASEGCN [23]	81.61	73.42	76.49	72.64	73.84	72.42
GL-GCN [36]	82.11	73.46	76.91	72.76	73.26	71.26
SK-GCN1 + BERT [42]	81.87	73.42	79.31	75.11	74.71	73.36
SK-GCN2 + BERT [42]	83.48	75.19	79.00	75.57	75.00	73.01
R-GAR + BERT [43]	86.60	81.35	78.21	74.07	76.15	74.88
CGAT + BERT [37]	86.25	80.38	80.41	76.48	77.46	76.37
DC-GCN (our)	88.11	83.28	81.77	77.09	77.84	76.77

5.5. Ablation Study

To further investigate the impact of each component of the DC-GCN model on the performance, we design several ablation experiments. The specific description of each component is as follows.

DC-GCN/P removes the position embedding layer P.

DC-GCN/S removes the syntactic graph convolution module S.

DC-GCN/M removes the multi-aspect sentiment graph convolution module M.

According to the data in Table 4, when the position embedding layer P is removed, the accuracy of DC-GCN on the three datasets is reduced by 0.55%, 1.34%, and 0.22%, respectively. This indicates that the position information between the aspect and the context word in the sentence is not negligible, especially in the Laptop dataset, and the contribution of the position information contributes significantly. Removing either S or M results in varying degrees of accuracy degradation, especially on the Restaurant dataset, and the accuracy and MF1 values drop significantly. This shows that a single GCN module cannot perform information interaction and thereby cannot learn more feature information between aspects and contexts. In general, DC-GCN incorporating all modules achieves the best performance.

Table 4. Ablation results of different components.

Model	Restaurant		Laptop		Twitter	
	Acc (%)	MF1 (%)	Acc (%)	MF1 (%)	Acc (%)	MF1 (%)
DC-GCN/P	87.56	82.98	80.43	75.85	77.62	76.29
DC-GCN/S	86.63	80.71	79.75	76.33	76.54	74.91
DC-GCN/M	86.39	80.45	79.89	76.56	76.96	75.08
DC-GCN	88.11	83.28	81.77	77.09	77.84	76.77

5.6. Case Study

For an intuitive understanding of the DC-GCN model and without the multi-aspect sentiment graph convolution (MAGCN) module, we take a multi-aspect example from each of the laptop and restaurant datasets as a case study. We visualize the attention scores of DC-GCN and DC-GCN (w/o MAGCN) in Table 5, where darker colors indicate higher attention scores for words.

Table 5. The visualization of attention scores.

Model	Aspect	Attention Visualization
w/o MAGCN	keyboard	I love the keyboard and the screen.
	appetizers	The appetizers are ok, but the service is slow.
DC-GCN	keyboard	I love the keyboard and the screen .
	appetizers	The appetizers are ok, but the service is slow.

The darker the color in the table the higher the attention score for the word.

For the first example “I love the keyboard and the screen”, with two aspects “keyboard” and “screen”, we can see that the w/o MAGCN model mainly focuses on the word “love” to predict the two aspects of polarity. For the DC-GCN model, in addition to the word “love”, it can also focus on the conjunction “and”. This result shows that DC-GCN, which fuses syntactic and multi-aspect sentiment graphs, can capture the sentiment dependencies of the two aspects through the conjunction “and”, and correctly predict the emotional polarity of the two aspects.

For the second example “The appetizers are ok, but the service is slow”, with two aspects “appetizers” and “service”, obviously, from the comma “,” and the conjunction “but”, we can know that the polarities of these two aspects are opposite. Although the w/o MAGCN model independently predicts the polarities of the two aspects, it ignores the relationship between the two aspects. In contrast, the DC-GCN model also pays attention to the conjunction “but” while predicting the polarity of both aspects.

The research case shows that our proposed DC-GCN model not only pays attention to the opinion words that help predict the sentiment of various aspects, but also considers the sentiment dependencies between different aspects in the sentence, which effectively improves the accuracy of aspect-level sentiment classification.

5.7. Impact of the DC-GCN Layer Number

To verify the impact of the number of DC-GCN layers on model performance, we compare the results using layers one to eight on the Restaurant and Laptop datasets. The experimental results are shown in Figure 5. From the data in the figure, we know that when the number of layers is 2, the performance of the model is the best. However, when the number of layers is less than 2 or more than 2, the accuracy and the MF1 value of the model are relatively lower. On the one hand, the feature learning between nodes is not sufficient when the number of layers is low. On the other hand, with the increase in the number of layers, the problem of gradient disappearance or gradient explosion easily occurs due to the increase in transformation operations in the GCN, which makes the model performance degenerate.

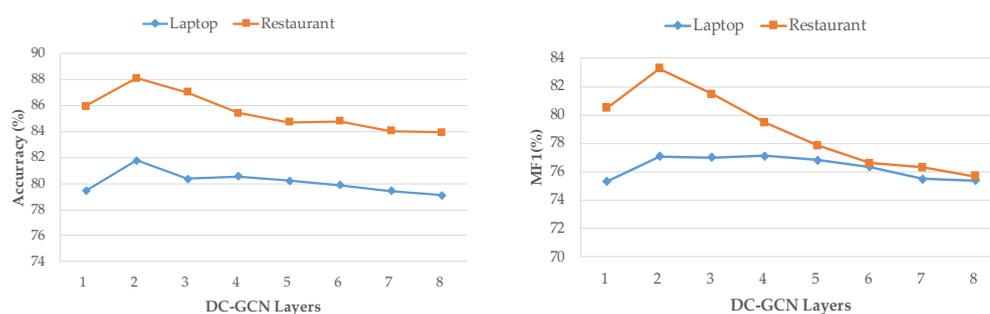


Figure 5. Effect of the number of DC-GCN layers.

6. Conclusions

In this paper, we propose a dual-channel interactive graph convolutional network (DC-GCN) model for aspect-level sentiment analysis. This model adopts the pre-trained *BERT* model to initialize word embedding, which makes it better than common methods such as Word2vec or Glove. In addition, we simultaneously consider the syntactic structure information and multi-aspect sentiment dependencies of sentences and employ two GCNs to learn their hidden representations separately, which is completely missing from previous research studies. Finally, we employ an attention mechanism to interactively learn syntactic and multi-aspect sentiment dependency information to obtain the final feature vector. The experimental results on three datasets show that the interactive learning of sentence syntactic information and multi-aspect sentiment dependency information can indeed effectively improve the performance of the model. In future work, we will discuss how to build a more accurate sentiment graph structure between aspects and explore how to introduce graph attention networks (GATs) into models to better model the relationship between aspects and opinion words.

Author Contributions: Conceptualization, Z.L. and Q.H.; methodology, Z.L. and Q.H.; software, Z.L.; validation, Z.L., Q.H. and L.Y.; formal analysis, Z.L.; investigation, Z.L.; resources, Q.H.; data curation, Z.L.; writing—original draft preparation, Z.L.; writing—review and editing, Z.L. and Q.H.; visualization, Z.L.; supervision, L.Y. All authors have read and agreed to the published version of the manuscript.

Funding: The research was funded by the National Natural Science Foundation of China “Research on the Evidence Chain Construction from the Analysis of the Investigation Documents (No.62166006)”, the National Natural Science Foundation of China “Rural spatial restructuring in poverty-stricken mountainous areas of Guizhou based on spatial equity: A case study of Dianqiangui Rocky Desertification Area (No.41861038)”, and Guizhou Provincial Science and Technology Projects (Guizhou Science Foundation-ZK [2021] General 335).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, B. Sentiment Analysis and Opinion Mining. In *Synthesis Lectures on Human Language Technologies*; Springer: Cham, Switzerland, 2012. Available online: <https://www.morganclaypool.com/doi/abs/10.2200/s00416ed1v01y201204hlt016> (accessed on 15 August 2022).
2. Schouten, K.; Frasincar, F. Survey on Aspect-Level Sentiment Analysis. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 813–830. <https://doi.org/10.1109/TKDE.2015.2485209>.
3. Ma, D.; Li, S.; Wu, F.; Xie, X.; Wang, H. Exploring Sequence-to-Sequence Learning in Aspect Term Extraction. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 3538–3547.

4. Dragoni, M.; Federici, M.; Rexha, A. An Unsupervised Aspect Extraction Strategy for Monitoring Real-Time Reviews Stream. *Inf. Processing Manag.* **2019**, *56*, 1103–1118. <https://doi.org/10.1016/j.ipm.2018.04.010>.
5. Wang, J.; Li, J.; Li, S.; Kang, Y.; Zhang, M.; Si, L.; Zhou, G. Aspect Sentiment Classification with Both Word-Level and Clause-Level Attention Networks. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018; pp. 4439–4445.
6. 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, Portland, OR, USA, 19–24 June 2011; pp. 151–160.
7. Weichselbraun, A.; Gindl, S.; Scharl, A. Extracting and Grounding Contextualized Sentiment Lexicons. *IEEE Intell. Syst.* **2013**, *28*, 39–46. <https://doi.org/10.1109/MIS.2013.41>.
8. Ding, X.; Liu, B.; Yu, P.S. A Holistic Lexicon-Based Approach to Opinion Mining. In Proceedings of International Conference on Web Search and Web Data Mining—WSDM '08, Palo Alto, CA, USA, 11–12 February 2008; ACM Press: New York, NY, USA, 2008; p. 231.
9. Gu, S.; Zhang, L.; Hou, Y.; Song, Y. A Position-Aware Bidirectional Attention Network for Aspect-Level Sentiment Analysis. In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, NM, USA, 21–25 August 2018; pp. 774–784.
10. Fan, F.; Feng, Y.; Zhao, D. Multi-Grained Attention Network for Aspect-Level Sentiment Classification. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 3433–3442.
11. Huang, L.; Sun, X.; Li, S.; Zhang, L.; Wang, H. Syntax-Aware Graph Attention Network for Aspect-Level Sentiment Classification. In Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain, 8–13 December 2020; pp. 799–810.
12. Ke, W.; Gao, J.; Shen, H.; Cheng, X. Incorporating Explicit Syntactic Dependency for Aspect Level Sentiment Classification. *Neurocomputing* **2021**, *456*, 394–406. <https://doi.org/10.1016/j.neucom.2021.05.078>.
13. Tran, T.T.; Miwa, M.; Ananiadou, S. Syntactically-Informed Word Representations from Graph Neural Network. *Neurocomputing* **2020**, *413*, 431–443. <https://doi.org/10.1016/j.neucom.2020.06.070>.
14. Asada, M.; Miwa, M.; Sasaki, Y. Extracting Drug-Drug Interactions with Attention CNNs. In Proceedings of the BioNLP 2017, Vancouver, BC, Canada, 4 August 2017; pp. 9–18.
15. Asada, M.; Gunasekaran, N.; Miwa, M.; Sasaki, Y. Representing a Heterogeneous Pharmaceutical Knowledge-Graph with Textual Information. *Front. Res. Metr. Anal.* **2021**, *6*, 670206. <https://doi.org/10.3389/frma.2021.670206>.
16. Su, J.; Tang, J.; Jiang, H.; Lu, Z.; Ge, Y.; Song, L.; Xiong, D.; Sun, L.; Luo, J. Enhanced Aspect-Based Sentiment Analysis Models with Progressive Self-Supervised Attention Learning. *Artif. Intell.* **2021**, *296*, 103477. <https://doi.org/10.1016/j.artint.2021.103477>.
17. Wu, Z.; Li, Y.; Liao, J.; Li, D.; Li, X.; Wang, S. Aspect-Context Interactive Attention Representation for Aspect-Level Sentiment Classification. *IEEE Access* **2020**, *8*, 29238–29248. <https://doi.org/10.1109/ACCESS.2020.2972697>.
18. Phan, H.T.; Nguyen, N.T.; Hwang, D. Convolutional Attention Neural Network over Graph Structures for Improving the Performance of Aspect-Level Sentiment Analysis. *Inf. Sci.* **2022**, *589*, 416–439. <https://doi.org/10.1016/j.ins.2021.12.127>.
19. Lu, G.; Li, J.; Wei, J. Aspect Sentiment Analysis with Heterogeneous Graph Neural Networks. *Inf. Processing Manag.* **2022**, *59*, 102953. <https://doi.org/10.1016/j.ipm.2022.102953>.
20. Zhang, Z.; Zhou, Z.; Wang, Y. SSEGCN: Syntactic and Semantic Enhanced Graph Convolutional Network for Aspect-Based Sentiment Analysis. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Seattle, WA, USA, 10–15 July 2022; pp. 4916–4925.
21. 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, Copenhagen, Denmark, 7–11 September 2017; pp. 452–461.
22. 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 (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 4567–4577.
23. Xiao, Y.; Zhou, G. Syntactic Edge-Enhanced Graph Convolutional Networks for Aspect-Level Sentiment Classification with Interactive Attention. *IEEE Access* **2020**, *8*, 157068–157080. <https://doi.org/10.1109/ACCESS.2020.3019277>.
24. Rao, D.; Ravichandran, D. Semi-Supervised Polarity Lexicon Induction. In Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009), Athens, Greece, 30 March–3 April 2009; pp. 675–682.
25. Ruder, S.; Ghaffari, P.; Breslin, J.G. A Hierarchical Model of Reviews for Aspect-Based Sentiment Analysis. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016.
26. Li, P.; Zhao, F.; Li, Y.; Zhu, Z. Law Text Classification Using Semi-Supervised Convolutional Neural Networks. In Proceedings of the 2018 Chinese Control And Decision Conference (CCDC), Shenyang, China, 9–11 June 2018; pp. 309–313.
27. Lv, Y.; Wei, F.; Cao, L.; Peng, S.; Niu, J.; Yu, S.; Wang, C. Aspect-Level Sentiment Analysis Using Context and Aspect Memory Network. *Neurocomputing* **2021**, *428*, 195–205. <https://doi.org/10.1016/j.neucom.2020.11.049>.
28. Dong, L.; Wei, F.; Tan, C.; Tang, D.; Zhou, M.; Xu, K. Adaptive Recursive Neural Network for Target-Dependent Twitter Sentiment Classification (Volume 2: Short Papers). In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, 22–27 June 2014; pp. 49–54.

29. Xue, W.; Li, T. Aspect Based Sentiment Analysis with Gated Convolutional Networks. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 15–20 July 2018; pp. 2514–2523.
30. Tang, D.; Qin, B.; Liu, T. Aspect Level Sentiment Classification with Deep Memory Network. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 214–224.
31. Tang, D.; Qin, B.; Feng, X.; Liu, T. Effective LSTMs for Target-Dependent Sentiment Classification. In Proceedings of the COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, Osaka, Japan, 11–16 December 2016; pp. 3298–3307.
32. 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, Austin, TX, USA, 1–5 November 2016; pp. 606–615.
33. Ma, D.; Li, S.; Zhang, X.; Wang, H. Interactive Attention Networks for Aspect-Level Sentiment Classification. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; pp. 4068–4074.
34. Huang, B.; Ou, Y.; Carley, K.M. Aspect Level Sentiment Classification with Attention-over-Attention Neural Networks. In Proceedings of the Social, Cultural, and Behavioral Modeling, Washington, DC, USA, 10–13 July 2018; Thomson, R., Dancy, C., Hyder, A., Bisgin, H., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 197–206.
35. Zhao, P.; Hou, L.; Wu, O. Modeling Sentiment Dependencies with Graph Convolutional Networks for Aspect-Level Sentiment Classification. *Knowl. Based Syst.* **2020**, *193*, 105443. <https://doi.org/10.1016/j.knosys.2019.105443>.
36. Zhu, X.; Zhu, L.; Guo, J.; Liang, S.; Dietze, S. GL-GCN: Global and Local Dependency Guided Graph Convolutional Networks for Aspect-Based Sentiment Classification. *Expert Syst. Appl.* **2021**, *186*, 115712. <https://doi.org/10.1016/j.eswa.2021.115712>.
37. Miao, Y.; Luo, R.; Zhu, L.; Liu, T.; Zhang, W.; Cai, G.; Zhou, M. Contextual Graph Attention Network for Aspect-Level Sentiment Classification. *Mathematics* **2022**, *10*, 2473. <https://doi.org/10.3390/math10142473>.
38. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv* **2016**, arXiv:1609.02907.
39. 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 Technology, Minneapolis, MN, USA, 2–7 June 2019.
40. Pontiki, M.; Galanis, D.; Pavlopoulos, J.; Papageorgiou, H.; Androutsopoulos, I.; Manandhar, S. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, 23–24 August 2014; pp. 27–35.
41. Song, Y.; Wang, J.; Jiang, T.; Liu, Z.; Rao, Y. Attentional Encoder Network for Targeted Sentiment Classification. In *Lecture Notes in Computer Science*; Springer: Cham, Switzerland, 2019; Volume 11730, pp. 93–103.
42. Zhou, J.; Huang, J.X.; Hu, Q.V.; He, L. SK-GCN: Modeling Syntax and Knowledge via Graph Convolutional Network for Aspect-Level Sentiment Classification. *Knowl. Based Syst.* **2020**, *205*, 106292. <https://doi.org/10.1016/j.knosys.2020.106292>.
43. 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, Online, 5–10 July 2020; pp. 3229–3238.