



# Aspect-Level Sentiment Analysis Using Dual Probability Graph Convolutional Networks (DP-GCN) Integrating Multi-Scale Information

---

Yunhui Pan, Dongyao Li, Zhouhao Dai and Peng Cui

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

October 29, 2023

# Aspect-based sentiment analysis using dual probability graph convolutional networks (DP-GCN) integrating multi-scale information

Yunhui Pan, Dongyao Li, Zhouhao Dai, and Peng Cui (✉)

Key Laboratory of Blockchain and Fintech of Department of Education of Guizhou, School of Information, Guizhou University of Finance and Economics, Guiyang, China  
{panyunhui121, lidongyao2021, z0815564}@163.com, pcui@mail.gufe.edu.cn

**Abstract:** Aspect-based sentiment analysis (ABSA) is a fine-grained entity-level sentiment analysis task that aims to identify the emotions associated with specific aspects or details within text. ABSA has been widely applied to various areas such as analyzing product reviews and monitoring public opinion on social media. In recent years, methods based on graph neural networks combined with syntactic information have achieved promising results in the task of ABSA. However, existing methods using syntactic dependency trees contain redundant information, and the relationships with identical weights do not reflect the importance of the aspect words and opinion words' dependencies. Moreover, ABSA is limited by issues such as short sentence length and informal expression. Therefore, this paper proposes a Double Probabilistic Graph Convolutional Network (DP-GCN) integrating multi-scale information to address the aforementioned issues. Firstly, the original dependency tree is reshaped through pruning, creating aspect-based syntactic dependency tree corresponding syntactic dependency weights. Next, two probability attention matrices are constructed based on both semantic and syntactic information. The semantic probability attention matrix represents the weighted directed graph of semantic correlations between words. Based on this, semantic information and syntactic dependency information are separately extracted via graph convolutional networks. Interactive attention is used to guide mutual learning between semantic information and syntactic dependency information, enabling full interaction and fusion of both types of information before finally carrying out sentiment polarity classification. Our model was tested on four public datasets, Restaurant, Laptop, Twitter and MAMS. The accuracy (ACC) and F1 score improved by 0.14% to 1.26% and 0.4% to 2.19%, respectively, indicating its outstanding performance.

**Keywords:** Aspect-based sentiment analysis; Graph neural network; Attention mechanism; Syntactic dependency tree.

## 1 Introduction

Sentiment analysis is an important research direction in the field of natural language processing, aimed at identifying the emotional bias in text. Compared to article-level sentiment analysis and sentence-level sentiment analysis, ABSA is a more fine-grained entity-level sentiment analysis task that aims to analyze and differentiate the emotional polarity expressed by different aspects in the same sentence. For example, in the sentence "The material of this clothes is very good but the price is expensive", "material" and "price" are aspect words of two aspects of the clothes. However, the emotional polarity of "material" is positive, and the emotional polarity of "price" is negative.

The key to the ABSA task is to establish a dependency relationship between all aspect words and their corresponding opinion words in the sentence, distinguishing each aspect word and its associated contextual information. In earlier research, Wang [1], Tang [2], Ma [3], Chen [4], and Fan [5] proposed various attention mechanisms to generate sentence representations specific to aspect words and model the relationship between aspect words and context words, achieving good results. For example, Wang [1] proposed an attention-based long short-term memory network for ABSA tasks, where the attention mechanism can focus on different parts of the sentence when different aspects are inputted. Tang [2] proposed a neural attention model that adds external memory to deep memory networks to capture the importance of each context word for inferring the emotional polarity of aspect words. Fan [5] proposed a multi-granularity attention network model (MGAN) to capture word-level interactions between aspect words and context words. However, models based on attention mechanisms are prone to mistakenly focusing on context information unrelated to aspect words, hence the attention mechanism is easily affected by additional information. Recently, with the development of graph neural networks (GNNs), using dependency parsers to parse the syntactic structure of sentences and generate syntactic dependency trees has gradually become a trend in solving ABSA tasks. Some researchers, such as Zhang [6], Liang [7], Wang [8], Li [9], have constructed different graph convolutional networks (GCNs) and graph attention networks (GATs), using the syntactic structure of sentences on the dependency tree to model the syntactic relationship between aspect words and context words. However, existing dependency trees not only contain a lot of redundant information but also assign the same weight to the dependency relationships of each edge in the sentence, resulting in a tree structure that neglects the importance of the dependency relationship between aspect words and their corresponding opinion words. In addition, some sentences with short lengths and informal expressions can cause models to perform poorly on data that is not sensitive to syntactic information.

In this paper, we propose a dual-probability graph convolutional network (DP-GCN) that combines multi-scale information to address the above two problems. For the first problem, we first obtain the original syntactic dependency tree of the sentence through the StanfordNLP parser, then reshape and prune the original tree to construct a syntactic dependency tree with aspect words as root nodes and with attached syntactic dependency weights. The syntactic dependency tree reshaped in this

way can not only clarify the syntactic dependency relationship between aspect words and their corresponding opinion words but also reveal the importance of the syntactic dependency information of individual words in the sentence with respect to aspect words. For the second problem, we extract and combine both linear structural semantic information and tree structural syntactic dependency information, respectively constructing probability attention matrices based on semantic and dependency information. We use graph convolutional networks to extract both semantic and syntactic dependency information, and then use an interactive attention mechanism to guide mutual learning between the two types of information.

The main contributions of this paper are as follows:

(1) We propose a dual-probability graph convolutional network (DP-GCN) that combines multi-scale information. We construct two probability attention matrices for semantic and syntactic dependency information, respectively, and send them into two graph convolutional networks. We utilize an interactive attention module to interactively learn semantic information and syntactic dependency information.

(2) We propose a syntactic dependency tree based on aspect words with attached dependency weights. The syntactic dependency weight reflects the importance of the syntactic dependency information of individual words in the sentence with respect to aspect words, making the syntactic dependency tree more suitable for ABSA.

(3) We conducted extensive experiments on the Restaurant dataset and Laptop dataset of SemEval2014 [24], Twitter dataset [25], and MAMS dataset [26] to evaluate our model, and the experimental results demonstrate the effectiveness of the DP-GCN model.

## 2 Related work

The key to the ABSA task is to establish the relationship between aspect words and their corresponding opinion words to distinguish the emotional tendencies corresponding to different aspect words in the same sentence. In earlier methods, feature vectors were usually designed manually and combined with machine learning algorithms to capture opinion words related to aspect words [10-13]. However, this approach cannot model the dependency relationship between aspect words and their context. Subsequently, various attention-based models [14-17] emerged, which implicitly model the semantic relationship between aspect words and context words to obtain opinion words corresponding to sentences and aspect words, and achieved good performance. Huang et al. [13] proposed an attention over-attention (AOA) network, which models both aspects and sentences jointly to capture interactions between aspects and contextual sentences. The AOA network learns representations of aspects and sentences together and automatically focuses on important parts of the sentences. Wang et al. [14] combined a multi-level interactive bidirectional gated recurrent unit (MI-bi-GRU), attention mechanism, and position features to allow their model to focus on target and contextual words that are important for sentiment analysis. Li et al. [15] proposed a hierarchical attention position-aware network (HAPN), which introduces positional embeddings to learn position-aware representations of

sentences and further generates target-specific representations of contextual words. Tan et al. [16] argued that expressing conflicting emotions towards an aspect (i.e., expressing both positive and negative emotions towards it simultaneously) is a common phenomenon. They suggested that excluding conflicting opinions is problematic and proposed a multi-label classification model with dual attention mechanism to address the issue of identifying conflicting opinions in existing models.

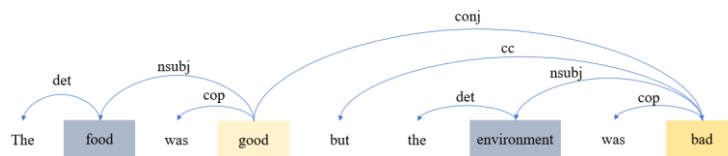
In addition, the pre-trained language model BERT [18] has achieved significant performance in natural language processing (NLP) tasks. Currently, many researchers [19-21] apply BERT pre-trained models to ABSA tasks, improving the performance of models in modeling semantic information of sentences, and better preparing for semantic interaction information between context and aspect words. For example, Sun et al. [19] construct an auxiliary sentence from the aspect and transform ABSA into a "sentence pair" classification task, and use fine-tuning BERT pre-trained models for ABSA tasks. Liang [21] proposed a bilingual syntax-aware graph attention network (BiSyn-GAT+), which fully utilizes the compositional tree information of a sentence's syntax (e.g., phrase segmentation and hierarchical structure) to simulate sentiment contexts of each aspect (intra-contexts) and cross-aspect sentiment relations (inter-contexts) for learning.

Currently, ABSA research mainly focuses on graph neural networks (GNNs) based on dependency trees. These methods explicitly utilize the syntactic structure information of sentences by extending graph convolutional network (GCN) and graph attention network (GAT) models through syntactic dependency trees, better handling the semantic and syntactic dependency relationships between aspect words and context, and proposing some outstanding models. For example, Zhang et al. [6] first applied GCN to ABSA, proposing a graph convolutional network on sentence dependency tree to solve the sentiment classification problem by utilizing dependency relationships in syntax information. Liang et al. [7] proposed an interactive graph convolutional network, identifying important aspect words and context words by constructing a heterogeneous graph for each sentence. Tang et al. [22] proposed a dependency graph enhanced dual Transformer network (DGEDT), which simultaneously considers both plane representation learned from Transformers and graph-based representation learned from corresponding dependency graph to iteratively model in an interactive manner. Specifically, DGEDT utilizes rich structural information by constructing a text sequence graph and an enhanced dependency graph, and designs a dual Transformer to model the structural information of the two graphs and learn sentence representations from two different perspectives. Wang et al. [8] created a unified aspect-oriented dependency tree structure, where the target aspect is the root node, by adjusting and refining a regular dependency parse tree. They proposed a relation graph attention network (R-GAT) to encode the new tree structure for sentiment prediction. Tian et al. [23] explicitly employed dependency types and used an attention mechanism to identify different types of dependencies. Li et al. [9] proposed a dual graph convolutional network model that simultaneously considered the complementarity of syntactic structures and the relationship of semantics.

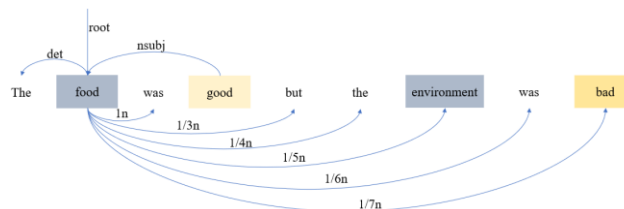
### 3 Reshaped syntactic dependency trees and multi-scale information

#### 3.1 Aspect-based syntactic dependency tree corresponding syntactic dependency weights

The syntactic dependency tree obtained by a regular syntactic parser contains the dependency relationships of all the words in the sentence, and all dependency relationships have the same weight. As shown in Figure 1, where there are many redundant dependency relationship types that are irrelevant to the ABSA task. However, the key to ABSA is to establish the relationship between aspect words and their opinion words. Therefore, reshaping and pruning the obtained syntactic dependency tree is necessary to obtain a syntactic dependency tree that is tailored to aspect words.



**Fig. 1.** Syntactic dependency tree including two aspect items, "food" and "environment", and two corresponding opinion words, "good" and "bad," in their context. The arrows in the figure indicate the dependency relationships between the two words, and the labels on the arrows represent the type of dependency relationship.



**Fig. 2.** In the figure, "food" is the root node, and all other dependency relationships are direct connections with "food".

Here are the steps to reshape and prune a regular syntactic dependency tree into a syntactic dependency tree based on aspect words and corresponding syntactic dependency weights: Firstly, we use a regular parser (StanfordNLP) to obtain the dependency tree of the input sentence. Then, we set the aspect word as the root node and generate a dependency tree based on the aspect word. If a sentence contains multiple aspects, then a tree based on each aspect will be constructed. Finally, the dependency tree is pruned so that words directly dependent on the aspect word have a dependency weight of 1 on their edge, the dependency weight of a word that does not have a direct dependency relationship with an aspect word is set to the reciprocal of its relative

position to the aspect word. Figure 2 shows the aspect-based syntactic dependency tree obtained after reshaping and pruning.

### 3.2 Multi-scale information

To address the lack of contextual information, this paper simultaneously uses linear-structured semantic information and tree-structured syntactic dependency information to reveal hidden information in the sentence.

**Positional distance.** In linear-structured sentences, the position and relative distance of each word in the sentence hold important information. By extracting the relative positional distances between each word and aspect words in the sentence, we can emphasize information from words closer to the aspect words and weaken information from words farther away from the aspect words. We can then use the positional distance to calculate the weights of each word in the sentence based on the aspect word. The calculation formula is as follows:

$$p_i = \begin{cases} 1 - (j_s - i)/n, & 0 \leq i \leq j_s \\ 0, & j_s \leq i \leq j_{s+m} \\ 1 - (i - j_{s+m})/n, & j_{s+m} \leq i \leq n \end{cases} \quad (1)$$

Here,  $p_i$  is the position weight of the  $i$ -th word,  $j_s$  and  $j_{s+m}$  are the start and end indexes of the aspect word.

**Dependency distance.** In the syntactic dependency information of a tree structure, dependency distance is the shortest distance between a word in a sentence and the aspect word in the syntactic dependency tree. Based on the dependency tree we constructed, the formula for constructing the dependency distance is shown below.

---

**Algorithm 1** Dependency distance algorithm based on aspect-based syntactic dependency tree

---

**Input:** index of aspect words (aspect\_idx), length of this sentence (n\_words), adjacency matrix (adj);

**Output:** distances: dependent distance sequence of each word based on the aspect word;

1: Use StanfordNLP for syntax analysis to get dependency tree and POS tags;

2: Reshape the dependency tree and transform it into a syntactic dependency tree based on aspect words.

3: distances = create an array of size n\_words, the initial value is -1

4:     Set distances[aspect\_idx] to 0

5:     Create an empty queue

6:     Add aspect\_idx to queue

7:     **When** the queue is not empty, execute the following steps in a loop:

8:         Take a node from the left side of the queue

9:         Traversing all neighbor nodes and corresponding weights of nodes in the adjacency matrix

10:             **If** weight  $\neq 0$  and distances[neighbor] = -1

11:                 distances[neighbor] = distances[node] + weight

12:             Add the neighbor to the queue

13:     **Return** distances

---

**Dependency relationship.** Dependency relationships can represent the syntactic relationships between words in the sentence's tree structure. If a word has a dependency relationship with the aspect word, then the corresponding edge in  $A_{rel}$  is set to the weight of the dependency for that word. If there is no dependency relationship, then the edge is set to 0. Thus,  $A_{rel}$  is constructed for the sentence, as shown in Figure 3.

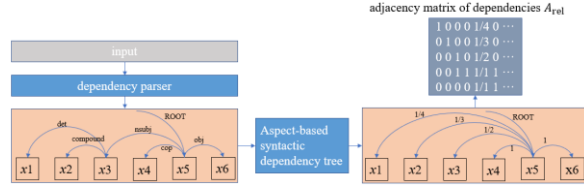


Fig. 3. Adjacency matrix  $A_{rel}$  of the dependency relationship.

**Dependency type.** The type of dependency relationship is a special and important piece of information. This paper first counts all dependency types in the dataset and generates a dependency types dictionary. Then, a randomly initialized vector for the initial dependency type corresponding to the text sequence  $S$  is generated, and a BiLSTM is used to obtain a feature vector  $h_{type} \in \mathbb{R}^{n \times d}$ , where  $n$  represents the length of the dependency type dictionary and  $d$  is the word vector dimension of the dependency type. Dependency types are embedded, as shown in Figure 4.

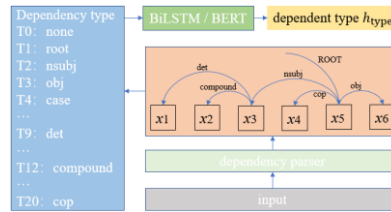


Fig. 4. Dependency type dictionary and dependency relationship types of the sentence.

## 4 Proposed DP-GCN model

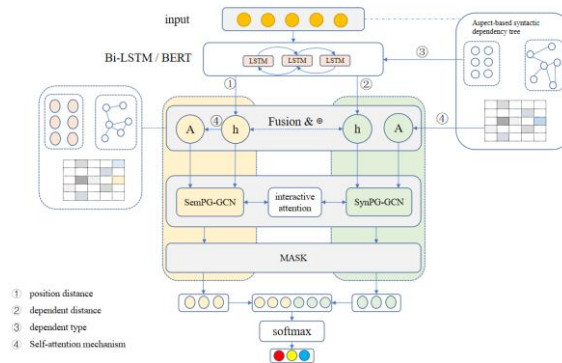


Fig. 5. Double probabilistic graph convolutional network (DP-GCN) integrating multi-scale information

Our proposed DP-GCN model is shown in Figure 5. In the ABSA task, given a sentence  $W^c = \{w_1^c, w_2^c, \dots, w_{r+1}^a, \dots, w_{r+m}^a, \dots, w_n^c\}$  containing  $n$  words, where  $W^a =$



$\{w_{r+1}^a, \dots, w_{r+m}^a\}$  is the aspect word sequence. Firstly, the words in the sentence are embedded into a low-dimensional vector space using an embedding matrix  $E \in \mathbb{R}^{|v| \times d_e}$ , where  $|v|$  is the vocabulary size and  $d$  represents the dimension of the word embedding. We use the StanfordNLP syntactic parser to parse the sentence and obtain its syntactic dependency information. Next, the obtained dependency type information is embedded into the low-dimensional vector space  $E \in \mathbb{R}^{|v| \times d_e}$ , where  $|v|$  is the size of the dependency type vocabulary and  $d$  is the dimension of the dependency type word embedding. Then, BiLSTM or BERT is used as the sentence encoder to extract the hidden contextual semantic representation  $h_{\text{sem}}$  and the dependency type representation  $h_{\text{type}}$ .

The hidden contextual semantic representation  $h_{\text{sem}}$  and the dependency type representation  $h_{\text{type}}$  of the sentence are fused with the multi-scale information. The fused representation  $h_{\text{input}}$  with multi-scale information is obtained by interacting the information through an interactive attention mechanism. Then,  $h_{\text{input}}$  is separately fed into the semantic probability graph convolutional module (SemPG-GCN) and the syntactic probability graph convolutional module (SynPG-GCN). Interacting attention is used to guide the communication of semantic information and syntactic dependency information during graph convolutions in both modules. Through masking, connection, and aggregation of aspect nodes, the final aspect representation is formed. Finally, sentiment polarity classification is performed using softmax. Next, we will describe the details of the DP-GCN model in detail.

#### 4.1 Interactive Attention.

The implementation of the interactive attention layer is mainly based on self-attention mechanism, which enables the model to simultaneously calculate the attention of contextual semantic features and dependency type features. Through the interactive attention mechanism, the dependency type features guide the learning of contextual features, while the contextual features guide the learning of dependency type features, as shown in Figure 6.

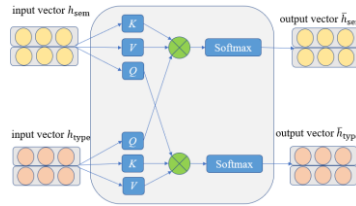


Fig. 6. Structure diagram of Interactive Attention.

#### 4.2 Fusion of Multi-scale Information.

This paper utilizes and integrates the multi-scale semantic information and multi-scale syntactic dependency information mentioned above as inputs to the model.

**Fusing contextual semantic information.** The positional distance is incorporated into the contextual representation as a weight parameter of the linear structure. This

context semantic information fused with the position distance can reflect the semantic association between different words and the aspect word in terms of distance. The fusion formula is as follows:

$$h_{\text{sem}} = F(h_{\text{sem}}) = p_i \cdot h_{\text{sem}} \quad (2)$$

Here,  $F$  is the positional weight function, and  $p_i$  is the positional weight of the  $i$ -th word. Thus, the closer the distance between words and the aspect word, the greater their relevance in the sentence, and the more significant their contribution to the judgment of sentiment polarity, since they have a higher weight value.

**Fusing syntactic dependency information.** Integrating dependency type and dependency distance information. The dependency distance reflects the importance of the syntactic dependency between each word in the sentence and the aspect word, which strengthens the words that have a direct syntactic dependency relationship with the aspect word and weakens those that do not have a direct relationship with the aspect word. The formula for fusing dependency type  $h_{\text{type}}$  with dependency distance is as follows:

$$h_{\text{type}} = F(h_{\text{type}}) = T * h_{\text{type}} \quad (3)$$

Here,  $F$  is the function for element-wise matrix multiplication,  $T$  is the dependency weight matrix composed of all dependency distances  $t_i$ . The multiplication is performed between the dependency type hidden vector and the corresponding element in the dependency weight matrix.

**Fusing semantic information and syntactic dependency information.** In this paper, interactive attention is used to fuse semantic information and syntactic dependency information, and the result of the fused information is used as the input to the model. Figure 6 shows the process of the interactive attention between the multi-scale semantic information  $h_{\text{sem}}$  and the multi-scale syntactic dependency information  $h_{\text{type}}$  to guide each other's learning.

Here, the multi-scale semantic information  $h_{\text{sem}}$  and the multi-scale syntactic dependency information  $h_{\text{type}}$  are used as inputs to the interactive attention. According to the Transformer model,  $h_{\text{sem}}$  and  $h_{\text{type}}$  are mapped to query ( $Q_{\text{sem}}$  and  $Q_{\text{type}}$ ), key ( $K_{\text{sem}}$  and  $K_{\text{type}}$ ), and value ( $V_{\text{sem}}$  and  $V_{\text{type}}$ ) matrices through linear layers. The formula for calculating  $h_{\text{sem}}$  using  $h_{\text{type}}$  is as follows:

$$C_{\text{sem}} = \text{softmax}\left(\frac{Q_{\text{type}}K_{\text{sem}}^T}{\sqrt{d}}\right)V_{\text{sem}} \quad (4)$$

$$\bar{h}_{\text{sem}} = \text{LN}(h_{\text{sem}} + C_{\text{sem}}) \quad (5)$$

Here, LN is a standardization function. Similarly,  $h_{\text{sem}}$  is used to guide  $h_{\text{type}}$ , as given in the following equation:

$$C_{\text{type}} = \text{softmax}\left(\frac{Q_{\text{sem}}K_{\text{type}}^T}{\sqrt{d}}\right)V_{\text{type}} \quad (6)$$

$$\bar{h}_{\text{type}} = \text{LN}(h_{\text{type}} + C_{\text{type}}) \quad (7)$$

Here,  $\bar{h}_{\text{sem}} \in \mathbb{R}^{n \times d}$  and  $\bar{h}_{\text{type}} \in \mathbb{R}^{n \times d}$  are both outputs of the interactive attention, and they use each other's feature information to enhance their own hidden representation abilities. Finally, the concatenated representation of the interactive semantic and syntactic dependency information is used as the input to the model, as shown in the following equation:

$$h_{\text{input}} = \bar{h}_{\text{sem}} \oplus \bar{h}_{\text{type}} \quad (8)$$

### 4.3 Semantic Probabilistic Graph Convolution Module.

**Semantic Probabilistic Graph Convolution (SemPG-GCN).** In order to fully focus the DP-GCN model on the aspect words and the corresponding opinion words, we use the self-attention mechanism to construct a probabilistic attention matrix  $A_{\text{sem}}$  about the multi-scale contextual semantic hidden representation  $h_{\text{sem}}$ , which is used as the input to the graph convolution. The specific formula is as follows:

$$A_{\text{sem}} = \text{softmax}\left(\frac{QW^Q \times (KW^K)^T}{\sqrt{d}}\right) \quad (9)$$

Here,  $Q$  and  $K$  are both the multi-scale contextual semantic hidden representation  $h_{\text{sem}}$ , while  $W^Q$  and  $W^K$  are learnable weight matrices, and  $d$  is the dimension of the multi-scale contextual semantic hidden representation  $h_{\text{sem}}$ .

Then, learn semantic information through graph convolutional networks. The specific formula of graph convolution is as follows:

$$h_i^l = \sigma\left(\sum_{j=1}^n A_{ij} W^l h_j^{l-1} + b^l\right) \quad (10)$$

Where  $h_i^l$  represents the hidden representation of node  $i$  in layer  $l$ , the initial value of the first layer is  $h_{\text{input}}$ .  $A_{ij}$  represents the element value in the  $i$ -th row and  $j$ -th column of matrix  $A_{\text{sem}}$ .  $W^l$  is a learnable parameter matrix,  $h_j^{l-1}$  is the hidden representation of neighboring nodes of  $h^l$  in layer  $l-1$ , and  $b^l$  is the bias term of the graph convolution.

### 4.4 Syntactic probabilistic graph convolutional module.

**Syntactic Probabilistic Graph Convolutional Networks (SynPG-GCN).** Nodes that have no dependency relationship with the aspect word are assigned 0, resulting in many zero elements in the generated adjacency matrix, which leads to the problem of missing information. The self-attention mechanism is applied to the matrix to obtain a continuous 0-1 probability matrix, which makes the model more robust and advanced.

$$A_{\text{rela}} = \text{softmax}(A_{\text{rel}} * W * U^T) \quad (11)$$

Where  $W$  and  $U$  are learnable weight matrices,  $A_{\text{rela}}$  is the probabilistic attention matrix of syntactic information, and  $A_{\text{rel}}$  is the adjacency matrix of the dependency relationship.

Then, learn syntactic dependency information through graph convolutional networks. The specific formula of graph convolution is as follows:

$$h_i^l = \sigma(\sum_{j=1}^n A_{ij} W^l h_j^{l-1} + b^l) \quad (12)$$

Where  $h_i^l$  represents the hidden representation of node  $i$  in layer  $l$ , the initial value of the first layer is  $h_{\text{input}}$ ,  $A_{ij}$  represents the element value in the  $i$ -th row and  $j$ -th column of matrix  $A_{\text{rela}}$ ,  $W^l$  is a learnable parameter matrix,  $h_j^{l-1}$  is the hidden representation of neighboring nodes of  $h^l$  in layer  $l-1$ , and  $b^l$  is the bias term of the graph convolution.

#### 4.5 Sentiment classification.

$h_{\text{input}}$  obtains  $h_{\text{semPG}}$  and  $h_{\text{synPG}}$  through SemPG-GCN and SynPG-GCN. Then, it multiplies with the aspect word masking matrix to extract the corresponding parts of the aspect word. The mask operation obtains  $\bar{h}_{\text{semPG}}$  and  $\bar{h}_{\text{synPG}}$ . They are concatenated and sent to the softmax layer to calculate the probability distribution of the input text in positive, negative, and neutral sentiment. The specific operation is as follows:

$$M_{i,j} = \begin{cases} 1, & i = j = p \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

Where  $M_{i,j}$  represents the element value in the  $i$ -th row and  $j$ -th column of the mask matrix. If  $i = j = p$ , which means the current position is the corresponding position of the aspect word, then the corresponding element value is set to 1. Otherwise, the corresponding element value is set to 0.

$$\bar{h}_{\text{semPG}} = h_{\text{semPG}} * M \quad (14)$$

$$\bar{h}_{\text{synPG}} = h_{\text{synPG}} * M \quad (15)$$

$$h_{\text{out}} = [\bar{h}_{\text{semPG}}; \bar{h}_{\text{synPG}}] \quad (16)$$

The probability of  $h_{\text{out}}$  after softmax is:

$$P(a) = \text{softmax}(Wh_{\text{out}} + b) \quad (17)$$

Where  $W$  and  $b$  are both learnable parameters,  $p(a)$  is the emotion probability distribution of the aspect word. In the model training process, cross-entropy is used as the loss function, and its formula is:

$$J = -\frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K y_{i,k} \log(\hat{y}_{i,k}) \quad (18)$$

Where  $N$  denotes the number of samples,  $K$  denotes the number of classes,  $y_{i,k}$  is the true label of sample  $i$  belonging to class  $k$ , and  $\hat{y}_{i,k}$  is the predicted probability of the model that the sample  $i$  belongs to class  $k$ .

## 5 Experiments.

### 5.1 Dataset and Evaluation Criteria.

This paper verifies the effectiveness of the DP-GCN model by conducting experiments on four publicly available datasets, which are Laptop and Restaurant datasets from SemEval2014 [24], Twitter dataset [25], and MAMS dataset [26]. Each sample in these four datasets is annotated with a sentiment label of one or more aspect words in a sentence, and the sentiment labels have three classifications: Positive, Negative, and Neutral. The statistical data for the number of samples in each category of the dataset is shown in Table 1.

**Table 1.** Experiment Data Statistics

Dataset	Positive		Neutral		Negative	
	Train	Test	Train	Test	Train	Test
Laptop	994	341	464	169	870	128
Restaurant	2164	728	637	196	807	182
Twitter	1561	173	3127	346	1560	173
MAMS	3380	400	5042	607	2764	329

This experiment uses two evaluation metrics, accuracy (Acc) and macro-average F1 score (MF1), to evaluate the effectiveness of the DP-GCN model.

### 5.2 Parameter Setting.

In this experiment, the experimental parameters of Glove and Bert are set as follows for the four datasets. The specific experimental parameters are shown in Table 2.

**Table 2.** Experimental Hyperparameter Settings

Experimental parameters	Set value
Num-epoch	50
Batch-size	16
Number of GCN layers	2
Number of LSTM layers	1
Number of interaction attention layers	2
Dependency type embedding dimension	40
BiLSTM hidden layer dimension	50
GCN hidden layer dimension	50
Max-length	85
L2 regularization	$10^{-5}$
Adam learning rate	0.002
Input/BiLSTM/GCN dropout	0.7/0.1/0.1
Early-stopping	500

### 5.3 Baseline Methods.

To comprehensively evaluate the performance of our model (DP-GCN), we compared it with the following baseline models on the four datasets:

1 ATAE-LSTM: Weighted the output of LSTM based on the attention mechanism to extract emotional words and features.

2 IAN: Simultaneously considered information at both the word and sentence levels in the text, and calculated the text representation using an interactive way so that the model can better capture the relationship between words and sentences.

3 RAM: When calculating the sentiment polarity of each aspect, not only the information of that aspect is considered, but also the information of other aspects is taken into account. The memory vector of each aspect is matched with the current input word vector sequence to obtain the attention vector of that aspect.

4 CDT: Used convolution on the dependency tree model to learn sentence features representation.

5 R-GAT: Used bidirectional GAT as the basic model, and employed relation-aware graph attention mechanism to capture the relationship between words and better capture information in the text sequence.

6 DGEDT: This model is based on a dual-channel LSTM, combined with a dynamic graph augmentation mechanism, which enables the utilization of both sentiment-embedded information and semantic information present in the text.

7 DualGCN: Built two graph convolution modules to process semantic information and syntactic dependency information.

8 T-GCN+BERT: Proposed a method that utilized a type-aware graph convolutional network (T-GCN) to explicitly depend on the ABSA type. Attention was used in T-GCN to distinguish different edges in the graph.

9 R-GAT+BERT: Used the pre-trained model BERT as the encoder instead of BiLSTM.

### 5.4 Experimental Results and Analysis

We conducted a three-class ABSA experiment on the four datasets from Section 4.1. The experimental results are shown in Table 4. The results in Table 4 indicate that our model (DP-GCN) has achieved a certain degree of improvement in both ACC and F1-score on the four public datasets.

From the experimental results of our model and the baseline model, it can be found that the performance of the DP-GCN model is better than models that solely use attention mechanism to capture aspect words and contextual words for modeling, such as ATAE-LSTM, IAN, etc. This suggests that the attention mechanism may only consider the semantic information of the sentence and cannot effectively capture the syntactic dependency information corresponding to the opinion words related to the aspect words. When dealing with longer sentences where aspects words and opinion words have distant dependencies, it is difficult to effectively identify the relationship between them. Models that consider the multiple aspect features of a sentence, such as RAM and CDT, introduce additional syntactic dependency information on the basis of the attention mechanism. However, the attention mechanism is easily affected by additional noise, making it difficult for the model to handle both semantic information

and syntactic dependency information effectively. Models that use graph neural networks (such as R-GAT, DGEDT, DualGCN) can capture words with long-distance dependencies in the context, which can better establish the relationship between aspect words and their opinion words. However, when dealing with informal datasets such as Twitter, these models have some limitations and do not consider the role of semantic information and syntactic dependency information in identifying relationships.

The DP-GCN model achieved good results in terms of ACC and F1-score on the four public datasets, indicating that the fusion of multi-scale information in the model input has integrated more semantic and syntactic dependency information of the sentences. The probability graph convolution module combined with an interactive attention mechanism enables the model to fully consider the semantic and syntactic information of the sentences. The enhancement in the model's performance indicates that to some extent, the syntactic dependency tree constructed by our model can mitigate the issue of the attention mechanism being susceptible to disruptions from noise.

In addition, the overall performance of the DP-GCN+BERT model proposed in this paper is also better than R-GAT+BERT and T-GCN+BERT, further demonstrating that the probability attention matrix with weighted syntactic dependency tree, semantic information, and syntactic dependency information has good effects on downstream tasks. Compared with the Glove-based DP-GCN model, DP-GCN+BERT improved the ACC by 1.27%~2.85% and F1-score by -0.46%~2.61%, and achieved better results than the non-BERT models in Table 3.

**Table 3.** Experimental results of different models on four public datasets

Models	Restaurant		Laptop		Twitter		MAMS	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
ATAE-LSTM	77.20	-	68.70	-	-	-	-	-
IAN	78.60	-	72.10	-	-	-	-	-
RAM	80.23	70.80	74.49	71.35	69.36	67.30	-	-
CDT	82.30	74.02	77.19	72.99	74.66	73.66	-	-
R-GAT	83.30	76.08	77.42	73.76	75.57	73.82	-	-
DGEDT	83.90	75.10	76.80	72.30	74.80	73.40	-	-
DualGCN	84.27	78.08	78.48	74.74	75.92	74.29	-	-
Our DP-GCN	84.76	78.48	79.74	76.20	76.06	76.48	81.96	81.15
R-GAT+BERT	86.60	81.35	78.21	74.07	76.15	74.88	-	-
T-GCN+BERT	86.16	79.95	80.88	77.03	76.45	75.25	83.68	83.07
Our DP-GCN+BERT	87.31	81.09	81.01	77.96	76.80	76.02	84.85	83.49

## 5.5 Ablation Experiment

In order to further study the role of a certain module in DP-GCN model, we conducted extensive ablation experiments. The results are shown in Table 5, and the specific experiments are as follows:

(1) **w/o location distance.** Remove the location distance information of the model, that is, reduce the dependency degree of the position distance in the semantic infor-

mation. As shown in Table 4, on the Restaurant, Laptop and MAMS datasets, the ACC and F1-score have decreased to some extent after removing the position information, while on the Twitter dataset, there is little change in ACC and F1-score. This suggests that the position information of the words in the sentence has little effect on the model's performance in datasets containing a large number of informal expressions.

(2) **w/o dependent type.** Remove the dependence type information, and the input of the model have only semantic information without the syntactic dependency information. The ACC and F1-score on all four datasets have decreased after removing the dependent type information, indicating that the dependency type information in the sentences can supplement the semantic information to some extent, allowing the model to learn more effective information.

(3) **w/o dependent tree.** Remove the tree based on the aspect word corresponding syntactic dependency weight, use StanfordNLP to generate the syntactic dependency tree, and also remove the dependency distance but retain the dependency type. The ACC and F1-score have shown a significant decrease on all four datasets after removing the dependent tree, indicating that reshaping the syntactic dependency tree is effective for ABSA tasks, and also suggesting that the original syntactic dependency tree contains redundant information.

(4) **w/o SemPG-GCN.** Remove the semantic information graph convolution module, and the ACC and F1-score have significantly decreased on all four datasets, indicating that the graph convolution module of the semantic information is the core module of this model, and suggesting that semantic information is essential for ABSA tasks.

(5) **w/o SynPG-GCN.** Remove the syntactic information graph convolution module, and the ACC and F1-score have decreased on all four datasets. From the experimental results, it can be seen that the syntactic information graph convolution module can complement the semantic information graph convolution module to some extent, and jointly improve the performance of the model.

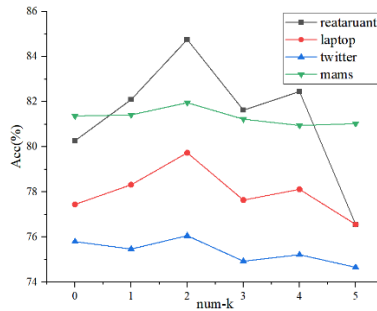
**Table 4.** Experimental results of ablation experiments

Models	Restaurant		Laptop		Twitter		MAMS	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
w/o location distance	83.41	76.25	78.63	75.21	75.95	76.31	81.06	80.26
w/o dependent type	82.53	73.14	76.48	73.21	74.63	73.69	80.22	80.32
w/o dependent tree	82.22	74.29	76.30	72.99	73.11	71.52	80.84	79.87
w/o SemPG-GCN	81.59	73.75	75.79	72.77	74.35	74.28	80.26	80.73
w/o SynPG-GCN	83.57	73.46	76.30	72.68	75.22	75.12	81.34	80.64

In summary, deleting distance information and dependency distance information will decrease the accuracy of our DP-GCN model, which illustrates the importance of the semantic information of the hidden linear structure and the syntactic information of the tree structure for the input information of the model. It can solve the problem of short sentences and informal expressions to some extent. Deleting the probability



attention matrix constructed by the self-attention mechanism of the SynPG-GCN module also leads to a decrease in accuracy, indicating that constructing a probability matrix about syntactic information through attention mechanisms can alleviate the influence of dependency parsing errors. Compared to comments from Restaurant and Laptop datasets, comments from Twitter are largely informal and insensitive to grammar information. Finally, the dependency tree and probability graph convolutional network that are based on aspect words and weighted dependencies are better suited for the MAMS dataset with multiple aspect words, as the relationship modeling between aspect words and corresponding opinion words becomes increasingly reliant on syntactic information as sentence complexity rises.



**Fig. 7.** The impact of the number of interactive attention layers on the model

Interactive attention is a critical module for the exchange of semantic information and syntactic dependency information. To explore the impact of the number of interactive attention layers on model performance, we investigated the number of interactive attention layers by setting the number of layers  $\text{num-k} = \{0, 1, 2, 3, 4, 5\}$ , respectively, and obtained the accuracy (ACC) of the four datasets, as shown in Figure 7.

As shown in Figure 7, the impact of the number of interactive attention layers on the model is nonlinear, and too few or too many layers can affect the performance of the model. In this experiment, when the number of interactive attention layers was 2, the highest accuracy was achieved in all four datasets. This may be because the interactive attention introduces different levels of interaction information while maintaining the consistency of the input feature space, which has a positive effect on improving the model performance. However, too many layers of interaction may introduce too much noise, leading to a decrease in model performance. Therefore, to obtain better performance in practical applications, it is necessary to adjust the number of interactive attention layers according to the specific dataset and task.

## 6 Conclusion

In this paper, we aimed to address the issue of redundant information in the current syntactic dependency trees for ABSA tasks. We proposed a tree structure based on aspect words corresponding syntactic dependency weights to systematically process

ABSAs tasks. We also proposed a dual probability graph convolutional network (DP-GCN) that combines multiscale information, which constructs two probability attention matrices to accommodate unclear or insignificant syntax and context semantic information. We used the interactive attention mechanism to guide the mutual learning of semantic and syntactic dependency information, thereby enhancing model expressiveness. Experimental results on datasets indicate that our DP-GCN model outperforms baseline models. However, our model still has limitations when processing datasets with many informal and biased expressions, such as the Twitter dataset. In future work, we will consider extracting other useful information related to semantic and syntactic information and optimizing the fusion of these two types of information. Additionally, we will improve the graph convolutional network model to enhance its generalization performance for ABSAs tasks.

**Acknowledgment.** This research was supported by the Top Talent Project of Guizhou Provincial Department of Education (QJJ-[2022]-080) and the Research Project for Students at Guizhou University of Finance and Economics (2022ZXSX163)

## References

1. Wang, Y., Huang, M., Zhu, X., Zhao, L.: Attention-based LSTM for Aspect-level Sentiment Classification. In: Conference on Empirical Methods in Natural Language Processing, (2016).
2. Tang, D., Qin, B., Liu, T.: Aspect Level Sentiment Classification with Deep Memory Network. arXiv preprint, arXiv:1605.08900, (2016).
3. Ma, D., Li, S., Zhang, X., Wang, H.: Interactive Attention Networks for Aspect-Level Sentiment Classification. arXiv preprint, arXiv:1709.00893, (2017).
4. Chen, P., Sun, Z., Bing, L., Yang, W.: Recurrent Attention Network on Memory for Aspect Sentiment Analysis. In: Conference on Empirical Methods in Natural Language Processing, (2017).
5. Fan, F., Feng, Y., Zhao, D.: Multi-grained Attention Network for Aspect-Level Sentiment Classification. In: Conference on Empirical Methods in Natural Language Processing, (2018).
6. Zhang, C., Li, Q., Song, D.: Aspect-based Sentiment Classification with Aspect-specific Graph Convolutional Networks. arXiv preprint, arXiv:1909.03477, (2019).
7. Liang, B., Yin, R., Gui, L., Du, J., Xu, R.: Jointly Learning Aspect-Focused and Inter-Aspect Relations with Graph Convolutional Networks for Aspect Sentiment Analysis. In: International Conference on Computational Linguistics, (2020).
8. Wang, K., Shen, W., Yang, Y., Quan, X., Wang, R.: Relational Graph Attention Network for Aspect-based Sentiment Analysis. In: Annual Meeting of the Association for Computational Linguistics, (2020).
9. Li, R., Chen, H., Feng, F., Ma, Z., Wang, X., Hovy, E. H.: Dual Graph Convolutional Networks for Aspect-based Sentiment Analysis. In: Annual Meeting of the Association for Computational Linguistics, (2021).
10. Titov, I., McDonald, R. T.: Modeling online reviews with multi-grain topic models. arXiv preprint, arXiv:0801.1063, (2008).
11. Ding, X., Zhang, Y., Liu, T., Duan, J.: Deep Learning for Event-Driven Stock Prediction. In: International Joint Conference on Artificial Intelligence, (2015).

12. Kiritchenko, S., Zhu, X.-D., Cherry, C., Mohammad, S. M.: NRC-Canada-2014: Detecting Aspects and Sentiment in Customer Reviews. In: International Workshop on Semantic Evaluation, (2014).
13. Huang, B., Ou, Y., Carley, K. M.: Aspect Level Sentiment Classification with Attention-over-Attention Neural Networks. arXiv preprint, arXiv:1804.06536, (2018).
14. Wang, X., Chen, X., Tang, M., Yang, T., Wang, Z.: Aspect-Level Sentiment Analysis Based on Position Features Using Multilevel Interactive Bidirectional GRU and Attention Mechanism. *Discrete Dynamics in Nature and Society*, vol. 2020, pp. 1-13, (2020).
15. Li, L., Liu, Y., Zhou, A.: Hierarchical Attention Based Position-Aware Network for Aspect-Level Sentiment Analysis. In: Conference on Computational Natural Language Learning, (2018).
16. Tan, X., Cai, Y., Zhu, C.: Recognizing Conflict Opinions in Aspect-level Sentiment Classification with Dual Attention Networks. In: Conference on Empirical Methods in Natural Language Processing, (2019).
17. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint, arXiv:1810.04805, (2019).
18. Sun, C., Huang, L., Qiu, X.: Utilizing BERT for Aspect-Based Sentiment Analysis via Constructing Auxiliary Sentence. In: North American Chapter of the Association for Computational Linguistics, (2019).
19. Xu, H., Liu, B., Shu, L., Yu, P. S.: BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis. arXiv preprint, arXiv:1904.02232, (2019).
20. Liang, S., Wei, W., Mao, X.-L., Wang, F., He, Z.: BiSyn-GAT+: Bi-Syntax Aware Graph Attention Network for Aspect-based Sentiment Analysis. In: Findings, (2022).
21. Tang, H., Ji, D.-H., Li, C., Zhou, Q.: Dependency Graph Enhanced Dual-transformer Structure for Aspect-based Sentiment Classification. In: Annual Meeting of the Association for Computational Linguistics, (2020).
22. Tian, Y., Chen, G., Song, Y.: Aspect-based Sentiment Analysis with Type-aware Graph Convolutional Networks and Layer Ensemble. In: North American Chapter of the Association for Computational Linguistics, (2021).
23. Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., Manandhar, S.: SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In: International Workshop on Semantic Evaluation, (2014).
24. Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., Xu, K.: Adaptive Recursive Neural Network for Target-dependent Twitter Sentiment Classification. In: Annual Meeting of the Association for Computational Linguistics, (2014).
25. Jiang, Q., Chen, L.-T., Xu, R., Ao, X., Yang, M.: A Challenge Dataset and Effective Models for Aspect-Based Sentiment Analysis. In: Conference on Empirical Methods in Natural Language Processing, (2019).
26. Sun, K., Zhang, R., Mensah, S., Mao, Y., Liu, X.: Aspect-Level Sentiment Analysis Via Convolution over Dependency Tree. In: Conference on Empirical Methods in Natural Language Processing, (2019).