Computing and Informatics, Vol. 44, 2025, 493–522, doi: 10.31577/cai_2025_3_493

TRIPLE-GCN: ENHANCED MULTI-FEATURE GRAPH CONVOLUTIONAL NETWORK FOR ASPECT-BASED SENTIMENT ANALYSIS

Huanling Tang

School of Computer Science and Technology
Shandong Technology and Business University, Yantai, China
e-mail: thl01@163.com

Xueyuan Sun

School of Information and Electronic Engineering
Shandong Technology and Business University, Yantai, China
e-mail: 395530561@qq.com

Quansheng Dou

School of Computer Science and Technology
Shandong Technology and Business University, Yantai, China
e-mail: li_dou@163.com

Mingyu Lu

School of Information Science and Technology
Dalian Maritime University, Dalian, China

Abstract. Aspect-Based Sentiment Analysis (ABSA) aims to predict the sentiment polarity of the given aspect word within the sentence. Recent studies frequently treat syntactic and semantic features as independent representations, thereby overlooking their intrinsic correlation. Concurrently, most of the existing methods largely neglect the significance of dependency types, which eventually impacts the accuracy of sentiment analysis. Research based on cognitive theory indicates a mutual influence between syntax and semantics. Based on this, we propose an ABSA model based on enhanced multi-feature graph convolutional network (Triple-GCN). Firstly, a shared enhanced graph convolutional module is proposed to integrate syntactic and semantic information. Following this, a thorough fusion of this syntactic and semantic information is carried out. Besides, relation and adjacency matrices are utilized for the innovative reconstruction of hidden state vectors. Syntactic graph convolution module dynamically fuses hidden state vectors and dependency features. Additionally, a position weight encoding function is designed to comprehend sentiment dependencies by drawing attention to aspect-near words. On the semantic side, dynamic semantic graphs are constructed, enabling the capture of semantic features. The model has been evaluated on three public datasets: Twitter, Laptop14, and Restaurant14. Compared to existing baseline models, the effectiveness of this model has noticeably improved.

Keywords: Aspect-based sentiment analysis, graph convolutional network, attention mechanism, dependency tree, common information, shared weight matrix

1 INTRODUCTION

Sentiment analysis is an important branch in natural language processing [1]. Early studies in sentiment analysis mainly regarded an article or a sentence as an information unit for overall sentiment analysis, but there may be multiple sentiment polarities in an article or a single sentence, which cannot satisfy the user's need to analyze the sentiment tendency of different aspects.

Aspect-Based Sentiment Analysis (ABSA) aims to identify the sentiment polarity of specific aspects in a given sentence [2, 3]. For instance, consider the review shown in Figure 1, which is taken from Restaurant14: "The price was reasonable although the service is poor." The comment contains two aspect words "price" and "service", the former corresponds to the positive affective word "reasonable" and the latter corresponds to the negative affective word "poor". The former corresponds to the positive sentiment word "good" and the latter corresponds to the negative sentiment word "poor".

The key of ABSA task lies in modeling the relationship between the aspect and its opinion words. Early ABSA tasks mainly utilized traditional machine learning methods, using manually extracted features, such as sentiment dictionaries and other tools for analysis [4, 5].

These methods require manual intervention and have poor generalization capabilities. With the continuous development of neural networks, they have been widely used in ABSA tasks. Previous studies have proposed various recurrent neural networks (RNNs) [6] with attention mechanisms to improved accuracy while

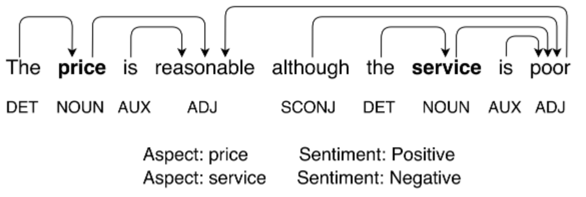

Figure 1. An example sentence with its dependency tree

analyzing the sentiment of less complex sentences. But these networks fell short when confronted with intricate sentences.

Incorporating an attention mechanism provides focused consideration on specific aspects [7, 8, 9]. However, the attention mechanism solely engages with context and aspect words at the semantic level, neglecting the syntactic relationship which exists between words.

This oversight could potentially lead to the misinterpretation of sentiment polarity. Syntactic information is mainly reflected in the dependency relationship between words, usually described by a dependency tree [10, 11, 12].

Recently, many studies have used graph convolutional networks (GCN) [13] or graph attention networks (GATs) [14] to encode syntactic, semantic, as well as common information of both [15, 16]. The distance between aspect and opinion words is shortened by the syntactic dependency tree, which effectively solves the problem of long-distance dependency. However, the incomplete syntactic structure of some sentences generates noise during the processing, resulting in unstable syntactic parsing results. In other words, the existing syntactic parsers are not specifically tailored for ABSA tasks. Therefore, methods that rely only on syntactic information also have shortcomings in ABSA tasks. Based on this, a dual-channel model based on syntax and semantics is proposed [17, 18]. In dual-channel models, syntactic and semantic information are processed in their own independent channels, and the results are simply concatenated for sentiment classification.

In the field of natural language processing, syntax and semantics are two key aspects. Syntax involves the structure and combination rules between words and phrases, while semantics involves the meaning and concept of words and phrases. Based on Pylkkänen, the study found that there is a certain degree of intersection and overlap between syntax and semantics [19, 20]. The posterior middle/superior temporal gyrus (pM/STG) that processes syntactic information intersects with the left anterior temporal lobe (LATL) and ventromedial prefrontal cortex (vmPFC) that process semantics. The intersection of pM/STG and LATL is shown in Figure 2. This means that the two brain regions pM/STG and LATL share functions when processing language. Specifically, the study conducted a comparison of the brain area activities in syntactic and semantic tasks, observed the activity patterns of these brain areas in different tasks, and concluded that it is difficult to distinguish between syntactic effects and semantic effects, while it found the intersection between pM/STG and LATL.

There are two limitations of existing ABSA methods:

1. In previous methods, syntactic and semantic information are usually processed in separate channels, and the extracted syntactic and semantic features are simply merged without fully utilizing the common information between them.
2. Some sentences lack obvious syntactic structure, which affects the accuracy of sentiment analysis.

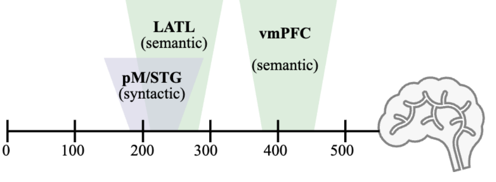

Figure 2. MEG results on processing stages of language comprehension

Based on these limitations, we propose an ABSA model based on enhanced multi-feature graph convolutional network (Triple-GCN). This model uses dependency syntactic analysis tools to construct a syntactic dependency graph, and employs a multi-head attention mechanism to construct a dynamic semantic graph to extract semantic information. Further, it constructs a shared enhancement module via a graph convolutional network to collate sentiment information from both syntactic and semantic aspects. The fusion of these information types enhances model performance. The main contributions of this paper are as follows:

- We propose a Triple-GCN model based on enhanced multi-feature graph convolutional network. This model integrates syntactic information with semantic information to emphasize the sentiment dependence between context words and aspect words.
- Combining human cognitive practice, this paper proposes a shared enhanced weight matrix to exchange features between syntax and semantics and learn similar feature representations at different levels. Besides, parameter sharing is implemented during the graph convolution process to reduce the amount of calculation.
- We design a dependency-based syntactic graph convolution (SynGCN) module to dynamically merge hidden state vectors and dependency features. And the position weight encoding function is proposed to comprehend sentiment dependencies by drawing attention to aspect-near words.
- Experiments were administered on three public datasets to ascertain the performance of Triple-GCN. The experimental results show that compared with the

most advanced ABSA model, the Triple-GCN model outperforms most mainstream baseline methods, establishing its proficiency in ABSA tasks. Furthermore, this paper analyzes the contribution of each module of the Triple-GCN model to its overall performance through ablation experiments, verifying the rationality and role of the model.

2 RELATED WORK

In recent years, the development of deep learning has given rise to numerous methods in ABSA. Various methods have been proposed to address the issues existing in ABSA, such as unstable syntactic parsing, insufficient utilization of syntactic and semantic information, and ignoring the connection between them. Currently, the main focus of existing ABSA methods is to model the correlation between a given aspect and its context. All of these methods can be classified into three primary categories:

1. semantic-based models;
2. syntactic-based models;
3. dual-channel models based on syntax and semantics.

2.1 Semantic-Based Models

Attention networks combined with deep neural networks can establish semantic relationships between themselves and their contexts. Therefore, most semantic-based models are developed on the basis of the attention mechanism. Wang et al. [11] proposed an Atention-based LSTM with Aspect Embedding, the model using attention can pay close attention to the upper and lower aspects of specific aspects. Ma et al. [21] proposed Interactive Attention Network for determining the attentional weights of the context. Fan and Feng [22] added a fine-grained attention mechanism to construct a multi-granular attention network model on this basis. Although the methods based on the attention mechanism achieved good results, due to the lack of utilization of syntactic information, the judgment of sentiment polarity was wrong when dealing with complex sentence structures or multiple aspects.

2.2 Syntactic-Based Models

By constructing a dependency tree through syntactic analysis, a syntax-based model is used to reduce the long-distance dependency between the subject and its opinion words. Zhang et al. [23] constructed an undirected graph based on the dependency tree and used GCN to learn contextual representations containing syntactic information. Huang and Carley [24] proposed a target dependency graph attention network and constructed a graph attention network based on the dependency tree for representation learning. Wang et al. [25] introduced dependency relationship information to construct a relationship graph attention network, but this method is overly dependent on the dependency tree. When the sentence structure is complex or the dependency tree parsing is wrong, the model performance will be affected. Sun et al. [26] stacked a GCN layer to extract rich representations over dependency tree. Liang et al. [27] build aspect-focused and inter-aspect graphs to learn aspect-specific sentiment features. Zhang and Qian [28] constructed a global lexical graph to capture the word co-occurrence relation and combined a global lexical graph and a syntactic graph.

2.3 Dual-Channel-Based Models

To solve above problems, the study tried to use a dual-channel model based on syntax and semantics to integrate syntactic and semantic information. Li et al. [17] constructed a dual-graph convolutional network based on syntax and semantics, extracted syntactic and semantic features respectively, and then realized the interaction between the two through a dual affine module, achieving good results. Pang et al. [16] built a semantic graph via multi-head self-attention mechanism. It takes dual-channel GCN to encode the sentence syntax and semantics, respectively.

3 THE APPROACH

The architecture of Triple-GCN is presented in Figure 3. The proposed model contains five major components:

1. Encoding layer;
2. GCN layers;
3. Masking layer;
4. Average-Pooling layer; and
5. Sentiment classification layer.

Details of each component are described as follows.

3.1 Encoding Layer

Given a sentence $s = \{w_1, w_2, \dots, w_{a_1}, w_{a_2}, \dots, w_{a_m}, \dots, w_n\}$ with n words, the aspect $a = \{w_{a_1}, w_{a_2}, \dots, w_{a_m}\}$ is a subsequence of sentence s. The aim of the Aspect-Based Sentiment Analysis (ABSA) task is to predict the sentiment polarity $y \in \{P, O, N\}$ for a given aspect, where P, O and N represent "Positive", "Neutral", and "Negative", respectively.

Each word in the sentence is embedded into a low-dimensional vector via a pre-trained word embedding matrix $\boldsymbol{E_e} \in \mathbb{R}^{|V_e| \times d_e}$, where $|V_e|$ is the size of the vocabulary list and d_e denotes the dimensionality of the word embedding, yielding the word embedding vectors $\boldsymbol{V} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_n\}$, where $\boldsymbol{v}_i \in \mathbb{R}^{d_e}$ signifies the word embedding

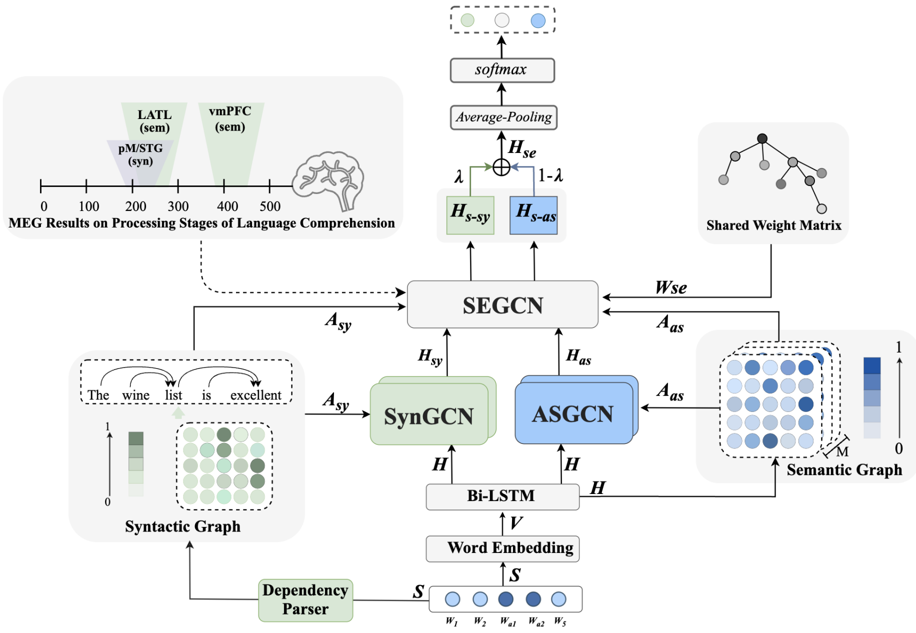

Figure 3. Triple-GCN model structure diagram

vector for the ith word. These embeddings are then fed into a BiLSTM layer, generating the hidden state vectors $\boldsymbol{H} = \{\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_n\}$, $\boldsymbol{H} \in \mathbb{R}^{d_{lstm} \times n}$, containing the contextual information between aspect words and opinion words, where d_{lstm} is the output dimension of the hidden state vector from the BiLSTM.

A sentence-aspect pair "$[CLS]$ sentence $[SEP]$ aspect $[SEP]$" is sent to BERT-base uncased [29] to obtain the hidden state vectors of the sentence. The BERT-base uncased model is a 12-layer transformer with 768 hidden units and 12 attention heads, pretrained on lower-cased English text. Then, the hidden representations of the sentence are input into the SynGCN and ASGCN, respectively.

3.2 Graph Convolutional Networks

Drawing inspiration from conventional convolutional neural networks (CNNs) and graph embedding, Graph Convolutional Networks (GCNs) are an efficient variant of CNNs that operate directly on graphs. GCN takes a graph as its input and generates an updated feature representation for each node within the graph as output. In handling graph-structured data, GCN can execute the convolution operation on immediately connected nodes, thereby encoding local information. Through the message-passing mechanism of multilayer GCNs, each node in a graph is capable of processing and learning from a wider range of information. For a graph containing

n nodes, we define the l^{th} layer graph structure as $G = (\boldsymbol{A}, \boldsymbol{H^{(l)}})$, where $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ acts as the adjacency matrix, $\boldsymbol{H^{(l)}}$ is the feature representation of the l^{th} layer of the GCN, and the computation of the $(l+1)^{\text{th}}$ layer of the GCN is shown as follows:

$$\boldsymbol{H^{(l)}} = \sigma\left(\boldsymbol{A}\boldsymbol{H^{(l-1)}}\boldsymbol{W^{(l)}} + \boldsymbol{b^{(l)}}\right), \tag{1}$$

where $\boldsymbol{W^{(l)}}$ is the learnable weight matrix, $\boldsymbol{b^{(l)}}$ is the bias, and $\sigma(\cdot)$ denotes the activation function.

In the research, we have incorporated three distinct graph convolution module, each contributing significantly to our work in unique ways. The SynGCN focuses on syntactic relations while the ASGCN handles aspects of semantic graphs. Lastly, the SEGCN serves as a shared enhanced graph convolutional network module for a more comprehensive understanding.

3.2.1 SynGCN

Inspired by [26], the Stanford parser is utilized to parse the sentence and obtain its syntactic dependency information. Meanwhile, The type of dependency and the distance between the aspect and its opinion word are also considered. Based on this, the Dependency-based Syntactic graph convolution (SynGCN) module is proposed. The dependency tree is transformed into the syntactic adjacency matrix $\boldsymbol{A_{ij}} \in \mathbb{R}^{n\times n}$ as follows:

$$\boldsymbol{A_{ij}} = \begin{cases} 1, & w_i, w_j \text{ contains dependencies,} \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Next, the obtained dependency type information is embedded into the low-dimensional vector space $\boldsymbol{E_r} \in \mathbb{R}^{|V_r|\times d_r}$, where $|V_r|$ is the size of the dependency type vocabulary and d_r is the dimension of the dependency type word embedding. Then, BiLSTM or BERT is used as the sentence encoder to extract the hidden state vectors $\boldsymbol{H}$ and the dependency type representation $\boldsymbol{H_{\text{type}}}$.

Combining $\boldsymbol{H_{\text{type}}}$ with the output $\boldsymbol{H}$ of the BiLSTM yields $H_{sy}^{(0)}$. With this operation, the model dynamically determines the extent to which semantic and dependency type information is used during the fusion process:

$$\boldsymbol{H_{sy}^{(0)}} = \alpha\boldsymbol{H} + (1-\alpha)\boldsymbol{H_{\text{type}}}, \tag{3}$$

where α is a hyper-parameter.

Specifically, there are L layers of SynGCN, $\boldsymbol{H_{sy}^{(0)}}$ is used as the initial input of the first SynGCN layer. When $l \in [1, L-1]$, the output of the $l-1$ layer before splicing is used as the input of the graph convolution of the l layer, and the graph convolution of the l layer is as follows:

$$H_{sy}^{(l)} = SynGCN\left(A_{sy}^{(l)}, H_{sy,in}^{(l}, W_{sy}^{(l)}\right), \tag{4}$$

where $\boldsymbol{A_{sy}^{(l)}}$ is the adjacency matrix of the l^{th} SynGCN layer, $\boldsymbol{H_{sy,in}^{(l)}}$ is the input of the l^{th} SynGCN layer, $\boldsymbol{W_{sy}^{(l)}} \in \mathbb{R}^{(d_{lstm}+(l-1)\times d_{gcn})\times d_{lstm}}$, d_{lstm} is the dimension of the hidden state vector, d_{gcn} is the output dimension of the graph convolution layer, and $\boldsymbol{H_{sy}^{(l)}}$ is the feature representation of the l^{th} SynGCN layer.

Words indicating sentiment tendencies are usually located near aspect words, and those more distant from aspect words generally contain less sentiment information. To enhance the importance of words neighboring to aspect words, a Position Weight (PW) encoding function was designed:

$$\boldsymbol{PW_i} = \begin{cases} 1 - \frac{i}{k}, & 0 \le i < k, \\ 0, & k \le i \le T, \end{cases} \tag{5}$$

where $\boldsymbol{PW_i}$ is the syntactic dependency distance of the i^{th} word, k represents the slope of the weight distribution function. Different k values correspond to different position weights, as shown in figure. T represents the predefined distance boundary.

The positional weights of $\boldsymbol{H_{sy}^{(L)}}$ are encoded using the syntactic dependency distance $\boldsymbol{PW}$, as shown in Equation (6):

$$\boldsymbol{H_{sy}^{(L)}} = \boldsymbol{PW}\boldsymbol{H_{sy}^{(L)}}, \tag{6}$$

where $\boldsymbol{PW}$ are positional weights used to reduce noise generated during dependency syntax analysis. The position-weighted $\boldsymbol{H_{sy}^{(L)}}$ is used as the final output of SynGCN.

3.2.2 ASGCN

The syntactic structure of some phrases is confusing, and their semantic information has a greater impact on the judgment of sentiment polarity. In order to obtain the deep semantic information, a dynamic semantic graph is designed, and the Attention-based Semantic graph convolution (ASGCN) module is proposed.

Specifically, the hidden state vector $\boldsymbol{H}$ is utilized as the first layer input of the L-layer ASGCN, with each layer's output of ASGCN serving as the succeeding layer's input. The M-head attention mechanism that ASGCN employs results in M attention score matrices (A_1, A_2,..., A_M), as demonstrated in Equation (7):

$$\boldsymbol{A}^{(l)} = \frac{\boldsymbol{H_{as,in}^{(l)}}\boldsymbol{W_{as,q}} \times (\boldsymbol{H_{as,in}^{(l)}}\boldsymbol{W_{as,k}})^{\boldsymbol{T}}}{\sqrt{\frac{d_{lstm}}{M}}}, \tag{7}$$

where $\boldsymbol{H_{as,in}^{(l)}}$ denotes the input of the l^{th} layer of ASGCN, $\boldsymbol{W_{as,q}}$ and $\boldsymbol{W_{as,k}}$ are the learnable weight matrices, d_{lstm} is the dimensionality of the hidden state vectors, and M is the number of attention heads.

Interpreting the attention score matrix as a fully connected graph allows irrelevant context words to contribute to feature extraction, which introduces noise. To address this issue, the *sorta* (sorted attention select) method is proposed, as shown in Figure 4.

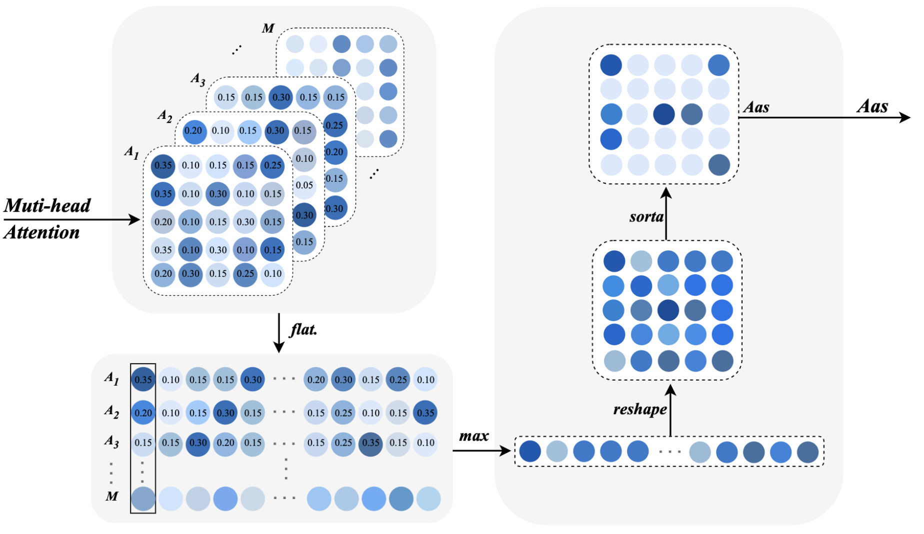

Figure 4. *sorta* attention selection method

The figure illustrates the *sorta* attention selection method used in our model, which plays a crucial role in refining the attention mechanism for aspect-based sentiment analysis. In the first stage, multiple attention score matrices (A_1, A_2, $\ldots$, A_M) are generated through the multi-head attention mechanism. These matrices represent how different parts of the input sequence influence each aspect word, with each head focusing on different contextual elements.

After generating these matrices, a softmax normalization is applied across each matrix to convert the attention scores into probabilities. This process ensures that the scores are comparable and interpretable as probabilities. Following this, the max function is used to select the highest probability values at each position across all matrices. This step is crucial because it identifies the most relevant attention heads, thereby highlighting the most significant contextual words for the current aspect word.

Next, using the *sorta* attention selection method, the most pertinent attention values corresponding to the top m context words are retained, and all other values are set to zero. This focused attention mechanism ensures that the model only considers the most relevant context when determining the sentiment of the aspect word. By reducing noise and focusing only on the most influential words, the model can better capture the sentiment nuances.

$$\boldsymbol{A'} = \max\left(\boldsymbol{A^{(1)}}, \ldots, \boldsymbol{A^{(M)}}\right), \tag{8}$$

$$\boldsymbol{A_{as}} = \text{sorta}(\boldsymbol{A'}), \tag{9}$$

where $\boldsymbol{A^{(i)}}$ denotes the i^{th} attention score matrix, $\boldsymbol{A_{as}}$ is the semantically most relevant attention score matrix, which serves as the neighbor matrix of ASGCN.

The l^{th} layer ASGCN is computed as shown in Equation (10):

$$\boldsymbol{H^{(l)}_{as}} = ASGCN(\boldsymbol{A^{(l)}_{as}}, \boldsymbol{H^{(l)}_{as,in}}, \boldsymbol{W^{(l)}_{as}}), \tag{10}$$

where $\boldsymbol{A^{(l)}_{as}}$ is the adjacency matrix of the l^{th} layer of ASGCN, the vector $\boldsymbol{H^{(l)}_{as,in}}$ serves as the input for the convolution process in the l^{th} layer of the graph. It represents the output from the $(l-1)^{\text{th}}$ layer, processed before splicing, ready to be utilized in the l^{th} layer, where $l \in [1, L-1]$.

The *sorta* method's ability to distill the attention to only the most critical words directly impacts the accuracy of sentiment classification, particularly in complex sentences where multiple sentiments might be present. Practically, this approach can be beneficial in applications like customer feedback analysis, where accurately identifying key sentiment drivers can lead to better insights and decision-making. Additionally, the effectiveness of this attention selection strategy could inspire future research in refining attention mechanisms for other natural language processing tasks.

3.2.3 SEGCN

As described by Pylkkänen [20], syntax and semantics are intrinsically linked, and common information needs to be highlighted in the feature extraction process to enhance sentiment transmission. To this end, a Shared Enhanced graph convolutional (SEGCN) module is designed, as shown in Figure 5.

In the SEGCN module, two separate Graph Convolutional Networks (GCNs) are used: one for syntactic information (SynGCN) and the other for semantic information (ASGCN). The syntactic graph $\boldsymbol{A_{sy}}$ and the semantic graph $\boldsymbol{A_{as}}$ serve as inputs to these GCNs, producing output features $\boldsymbol{H_{sy}}$ and $\boldsymbol{H_{as}}$, respectively. The shared weight matrix $\boldsymbol{W_{se}}$ then processes these features to generate new output features $\boldsymbol{H_{s-sy}}$ and $\boldsymbol{H_{s-as}}$, which are further combined using a parameter λ to form the final output $\boldsymbol{H_{se}}$:

$$\boldsymbol{H_{s-sy}} = SynGCN(\boldsymbol{A_{sy}}, \boldsymbol{H_{sy}}, \boldsymbol{W_{se}}), \tag{11}$$

$$\boldsymbol{H_{s-as}} = ASGCN(\boldsymbol{A_{as}}, \boldsymbol{H_{as}}, \boldsymbol{W_{se}}). \tag{12}$$

The final output of SEGCN $\boldsymbol{H_{se}}$ is as shown in the formula:

$$\boldsymbol{H_{se}} = \lambda \boldsymbol{H_{s-sy}} + (1-\lambda)\boldsymbol{H_{s-as}}, \tag{13}$$

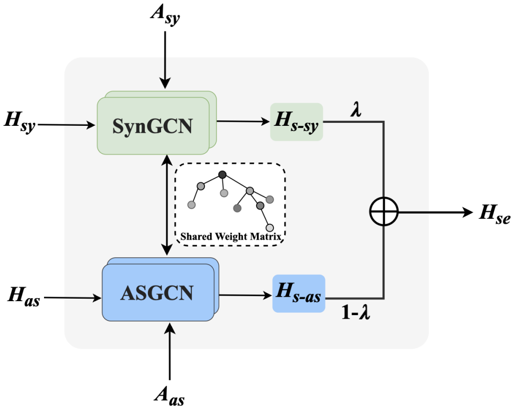

Figure 5. Shared Enhanced graph convolutional module

where λ is the weight coefficient used to dynamically compromise and fuse two feature representations with different semantics.

The SEGCN module's ability to share parameters between the syntactic and semantic spaces ensures that the most critical information is retained, thereby improving the model's understanding of complex language structures. This method is particularly effective in capturing the nuances of sentiment in text, where both syntax and semantics play crucial roles. The practical implications of this module extend to various natural language processing tasks, such as customer feedback analysis, where understanding the underlying sentiment is vital for decision-making. Additionally, this approach can inspire future research in developing more sophisticated models that better integrate multiple linguistic features for enhanced performance in sentiment analysis and related tasks.

3.3 Masking Layer and Average Pooling Layer

After the graph convolution layers, we obtain a shared feature representation $\boldsymbol{H_{se}} = \{\boldsymbol{h_{se,1}}, \ldots, \boldsymbol{h_{se,n}}\}$ that fully integrates syntactic and semantic information. In order to highlight the important features of aspect words, the output vector $\boldsymbol{H_{se}}$ of the final GCN layer is context-masked, and only the final representation of aspect words is retained:

$$\boldsymbol{h_t} = \begin{cases} 0, & 1 \leq t < a+1,\ a+m < t \leq n, \\ 1, & a+1 \leq t \leq a+m, \end{cases} \tag{14}$$

where t represents the position of the word in the text sequence, $\boldsymbol{h_t}$ represents the vector representation of the position t in the text sequence, and a represents the starting point of the aspect word in the text sequence The starting position, m, represents the length of aspect.

Perform mask operation on the output of the last layer of SEGCN:

$$\boldsymbol{H_{mask}} = \{0, \ldots, \boldsymbol{h_{a+1}}, \boldsymbol{h_{a+2}}, \ldots, \boldsymbol{h_{a+m}}, \ldots, 0\}, \tag{15}$$

where $\boldsymbol{H_{mask}}$ represents the masked output vector of the last layer of SEGCN, and $\boldsymbol{h_{a+m}}$ is the retained feature representation corresponding to the aspect word.

Perform an average pooling operation on the feature representation $\boldsymbol{H_{mask}}$ output by the mask layer:

$$\boldsymbol{H_p} = \mathrm{Average}(\boldsymbol{H_{mask}}), \tag{16}$$

where $\mathrm{Average}(\cdot)$ represents average pooling, and $\boldsymbol{H_p}$ is the final aspect feature representation.

3.4 Affective Classification Layer

In order to predict the sentiment polarity of a given aspect, the final aspect feature representation $\boldsymbol{H_p}$ is used as the input of the fully connected layer and the probability distribution y is obtained through the $softmax$ layer:

$$y = \mathit{softmax}(\boldsymbol{W}\boldsymbol{H_p} + b), \tag{17}$$

where $\boldsymbol{W}$ and b represent the trainable weight matrix and bias of the fully connected layer, respectively.

3.4.1 Model Training

The objective function of the model uses the cross entropy loss function and is trained using the standard gradient descent method:

$$\mathit{loss} = -\sum_{i=1}^{S}\sum_{j=1}^{C} y_i^j \log \hat{y}_i^j + \lambda||\theta||^2, \tag{18}$$

where S represents the size of the training set, i represents the i^{th} sample, C represents the number of sentiment polarity categories, j represents the j^{th} sentiment polarity, y is the true probability distribution, $\hat{y}$ is the predicted probability distribution, λ is the weight coefficient of L_2 regularization, and θ is the model parameter.

4 EXPERIMENTAL SETTINGS

This section introduces the benchmark datasets used for evaluation and the baseline models used for comparison. The experimental results are analyzed from different perspectives, including datasets, implementation and parameter settings, baseline models, comparative experiments, ablation experiments, sample analysis, hyperparameters, and the number of model layers.

4.1 Datasets

To verify the effectiveness of the Triple-GCN model, experiments were conducted on three widely used benchmark datasets: Twitter, Laptop14, and Restaurant14. These datasets are particularly suitable for aspect-based sentiment analysis tasks and provide a comprehensive evaluation of the model's performance.

The Twitter dataset, originally provided by Dong et al. [30] and later adapted by Zhang et al. [23], includes 6 051 training samples and 677 testing samples. It focuses on sentiment polarities in short texts from tweets, capturing the challenges of analyzing sentiment in unstructured social media environments where language is often informal and varied.

The Laptop14 and Restaurant14 datasets, both derived from the SemEval-2014 Task 4 [31], contain detailed aspect-specific sentiment annotations. These datasets are crucial for training and evaluating models that perform aspect-based sentiment classification. Laptop14 comprises 2 282 training samples and 632 testing samples from laptop reviews, making it particularly suitable for sentiment analysis in the consumer electronics domain. Restaurant14 consists of 3 608 training samples and 1 119 testing samples from restaurant reviews, providing insights into the model's robustness across diverse product review contexts. Additionally, following Chen et al. [7], instances labeled as "conflict" were removed to maintain dataset consistency. The consistent use of these datasets in research underscores their reliability and richness in annotated sentiment data.

The sample label distribution of the dataset is shown in Table 1. The column headers use abbreviations to represent specific aspects of dataset statistics relevant to sentiment analysis. The Datasets column indicates the name or identifier, such as Twitter, Laptop14, or Restaurant14. Samples refers to the total number of entries in each dataset subset (Training, Testing). Positive denotes Positive samples, Neutral stands for Neutral samples, and Negative indicates Negative samples. Understanding these terms is crucial for interpreting dataset statistics in sentiment analysis tasks, which in turn helps researchers and practitioners evaluate dataset quality, sentiment balance, and aspect diversity.

Datasets	Division	Samples	Positive	Neutral	Negative
Twitter	Training	6 051	1 507	3 016	1 528
	Testing	677	172	336	169
Laptop14	Training	2 282	976	455	851
	Testing	632	337	167	128
Restaurant14	Training	3 608	2 164	637	807
	Testing	1 119	727	196	196

Table 1. Statistics for the experimental datasets

4.2 Implementation Details

The experiment uses consistent implementation and parameter settings across different models, employing Stanford CoreNLP [32] as the dependency parser and 300-dimensional pre-trained GloVe embeddings [33] to initialize word embeddings. Additionally, part-of-speech embeddings and position embeddings are used to enrich sentence representations.

Specifically, part-of-speech embeddings encode the grammatical category of each word using a vector representation. Position embeddings encode the relative distance of aspect words within sentences to enhance aspect-based sentiment analysis. The dimensions of these two embedding vectors are set to 30. The word, part-of-speech, and position embeddings are concatenated as the output of the embedding layer, serving as input to the Bi-LSTM layer. To mitigate overfitting, a dropout parameter of 0.7 is applied to BiLSTM, SynGCN, ASGCN, and SEGCN layers. The hidden state vector from BiLSTM has a dimensionality of 100, and the Adam optimizer is employed with a learning rate of 1.0×10^{-3}.

The number of layers for SynGCN, ASGCN, and SEGCN is set to 3 for Laptop14 and Restaurant14 datasets because these datasets contain more complex syntactic structures that benefit from deeper layers to capture intricate relationships between words. For the Twitter dataset, the number of layers is set to 2 as the text is shorter and generally simpler in structure, making fewer layers sufficient for capturing relevant syntactic and semantic information.

For the BERT-based model, the BERT-base uncased version [29] is used, with a word embedding dimension of 768. The batch size is set to 16, and the learning rate is adjusted to 0.00002. Model performance is evaluated using Accuracy (Acc) and Macro-F1 (MF1) scores. Detailed parameter settings are available in the code.

4.3 Baseline Methods

In order to verify the effectiveness of the model, 11 related methods are selected for comparison:

- **ATAE-LSTM** [11]: The aspect vector is embedded into the word vector and hidden vector, so that the aspect information participates in the calculation of attention weight, which strengthens the performance of LSTM on the ABSA task.
- **MGAN** [22]: A multigrained attention mechanism is designed to capture word-level interactions between the aspect and context.
- **IAN** [21]: Using interactive attention to learn the association between different aspects and emotions in text, it can effectively capture the emotional information related to specific aspects in text, and can adjust its focus when considering different aspects.
- **IGATs** [34]: An interactive graph attention network model is proposed to capture dependency information, semantic relations and position information.

- **ASGCN** [23]: The syntactic dependency tree and dependency graph of the sentence are constructed and input into the graph convolutional network to model the syntactic relationship between aspect words and context.
- **CDT** [26]: Through the concatenation of BiLSTM and GCN, the information between aspect words and context in the text is extracted.
- **Bi-GCN** [28]: A two-layer interactive graph convolutional network is proposed to distinguish various types of dependency relations or word co-occurrence relations by building a concept hierarchy on the syntactic graph and semantic graph.
- **R-GAT** [25]: The dependency tree is reconstructed and GAT is applied for representation learning, removing redundant information and taking aspect words as the root nodes of the dependency tree.
- **DGEDT** [35]: Use GCN to process graph features, use Transformer to process plane features, and then fuse the feature representations.
- **DualGCN** [17]: Use SynGCN to capture syntactic information, while SemGCN is used to capture semantic information. Differential regularizers and orthogonal regularizers are then designed to improve model performance.
- **DMGCN** [16]: A multi-channel GCN method is designed to encode the syntax, semantics and correlated information from the generated graph.

4.4 Comparative Experiment

Type	Model	Twitter		Laptop14		Restaurant14	
		Acc.	MF1	Acc.	MF1	Acc.	MF1
Syn.	CDT	74.13	72.05	75.23	72.26	81.65	73.39
	R-GAT	75.21	73.97	76.27	73.32	82.91	75.52
	IGATs	75.07	73.84	77.42	73.47	82.51	76.25
	DGEDT	74.80	74.51	76.76	72.63	83.30	75.87
Att.	ATAE-LSTM	66.10	65.32	68.28	65.51	76.64	67.87
	IAN	72.12	70.12	72.29	70.03	79.15	70.27
	MGAN	72.95	70.81	75.29	72.03	80.15	73.27
GCN	ASGCN	72.30	70.81	74.42	71.36	80.93	73.54
	BiGCN	73.68	72.42	74.12	71.35	81.48	75.87
	DualGCN	77.50	75.15	77.87	**75.09**	82.21	78.08
	DMGCN	77.76	75.61	77.45	74.83	82.64	78.35
Ours	**Triple-GCN**	**78.53**	**76.25**	**79.03**	74.80	**83.52**	**78.44**
BERT	DGEDT-BERT	76.30	75.21	79.42	75.36	86.63	80.04
	DualGCN-BERT	77.48	77.02	80.12	77.35	87.19	81.47
	DMGCN-BERT	**77.74**	76.28	80.42	78.03	86.42	81.29
	Triple-GCN-BERT	77.50	**77.15**	**80.67**	**78.29**	**87.21**	**81.52**

Table 2. Comparative experiment

In this section, we evaluate ABSA models using two main evaluation metrics: Accuracy and Macro-averaged F1-score, across three established benchmark datasets. The comparative results, displayed in Table 2, show that the Triple-GCN model outperforms others, particularly on Twitter and Restaurant14, where it achieves the best Accuracy and MF1 scores. Although it performs slightly less effectively on Laptop14 compared to DualGCN, the overall performance underscores the model's capability in ABSA tasks.

The significant improvements in Accuracy and MF1 on Twitter, particularly by 3.73 % and 1.74 % over syntax-based models like DGEDT, indicate that Triple-GCN is well-suited for analyzing short and informal texts common in social media. The integration of semantic information proves crucial in enhancing sentiment analysis in such contexts, effectively capturing nuances and improving overall model accuracy.

On Laptop14 and Restaurant14, Triple-GCN shows improved performance over attention-based models like MGAN, with gains in Accuracy and MF1 up to 5.17 %. This indicates that the design of the model, which combines syntactic and semantic information via a shared weight matrix, effectively captures essential elements of sentiment analysis, providing robustness across different types of text inputs.

Compared to DMGCN, Triple-GCN further demonstrates its strength, particularly on Twitter, with an additional increase in Accuracy and MF1 by 0.77 % and 0.64 %, respectively. This improvement is attributed to the shared enhanced graph convolution module, which effectively integrates syntax and semantics, especially in less structured text environments, highlighting the model's adaptability in varied contexts.

Finally, the introduction of BERT-based models further boosts Triple-GCN's performance, especially on Laptop14 and Restaurant14, where Accuracy and MF1 saw modest gains. This underscores the robustness and flexibility of Triple-GCN when leveraging pre-trained language models for encoding, making it a strong contender for future sentiment analysis applications.

4.5 Ablation Experiments

In order to verify the impact of SynGCN, ASGCN, and SEGCN on the overall performance of Triple-GCN, ablation experiments were conducted to compare the performance of Triple-GCN and its variants on three public datasets, as shown in Table 3. The variants include GCN, w/o SynGCN, w/o ASGCN, and w/o SEGCN. GCN represents the result obtained by using only the graph convolution network, w/o represents the model without the corresponding module, SynGCN represents the syntactic graph convolution module, ASGCN represents the semantic graph convolution module based on the attention mechanism, SEGCN represents the shared enhanced graph convolution module, and Triple-GCN is the baseline model.

Table 3 clearly illustrates that the three graph convolution modules considerably enhance the model's performance. This is particularly evident in the significant Accuracy and MF1 improvements across all datasets when each module is included, demonstrating that these modules are crucial for capturing both syntactic and semantic information effectively.

Without the syntactic graph convolution module (w/o SynGCN), the model's proficiency on Laptop14 has notably diminished, as indicated by decreases in Accuracy and MF1 of 3.53 % and 3.58 %, respectively, and on Restaurant14 by 1.38 % and 1.87 %, respectively. This underlines the critical role syntactic information plays in sentiment analysis, particularly in domains where sentence structure heavily influences sentiment.

If the semantic graph convolution module is neglected (w/o ASGCN), the performance of the model markedly falls across three datasets. This drop is especially apparent within Twitter, with Accuracy and MF1 decreasing by 5.06 % and 3.74 %, respectively. This can be attributed to the non-standard syntax used in Twitter comments and their minimal reliance on syntactic information, emphasizing the importance of the ASGCN module in effectively extracting deep semantic data from less structured text environments.

Without the shared enhanced graph convolution module (w/o SEGCN), the performance of the model on Laptop14 notably declines, with Accuracy reducing by 2.84 % and MF1 by 1.68 %. This illustrates that the SEGCN module is critical in seamlessly amalgamating syntactic and semantic information, ensuring that the model adapts well to varying text structures and generates more precise feature representations. The results highlight the importance of integrating both syntax and semantics in emotion classification tasks.

Furthermore, replacing the graph convolution module with a traditional graph convolution network (GCN) leads to a significant performance drop, proving that the Triple-GCN structure proposed in this article is more effective in learning complex syntax and semantics. Through these ablation experiments, we confirmed the effectiveness of each component of the Triple-GCN model and demonstrated that its overall structure provides strong performance in aspect-level sentiment analysis tasks.

Models	Twitter		Laptop14		Restaurant14	
	Acc.	MF1	Acc.	MF1	Acc.	MF1
GCN	71.68	70.05	74.12	68.35	77.43	71.87
w/o SynGCN	75.18	73.82	76.50	73.22	82.14	75.27
w/o ASGCN	73.47	72.31	76.97	73.77	82.82	75.85
w/o SEGCN	76.56	75.32	76.19	73.12	82.76	76.14
Triple-GCN	**78.53**	**76.25**	**79.03**	**74.80**	**83.52**	**78.44**

Table 3. Comparison of experimental results

4.6 Sample Analysis

In order to explore the impact of syntactic graph convolution and semantic graph convolution on the Triple-GCN model in more detail, real samples were collected from the test set. GCN, SynGCN, ASGCN, and Triple-GCN were used to predict the sentiment polarity of the model, and the results are shown in Table 4. In the table, underscores are used to represent aspect words in the sentence, "*P*", "*O*", and "*N*" are used to indicate whether the model correctly predicts the sentiment polarity of the sample.

	Sample	Syn.	Sem.	Trip.
1	The **food** not worth the price. (N)	N(T)	N(T)	N(T)
2	The **settings** are not convenient either. (N)	N(T)	N(T)	N(T)
3	I thought that it will be fine, if I do some **settings**. (O)	O(T)	P(F)	O(T)

Table 4. Sample prediction results

For the sample 1, only GCN cannot make a correct prediction because "food" is syntactically closer to "worth" which represents positive sentiment, so GCN gives an incorrect prediction. The SynGCN module increases the weight of "not" and decreases the weight of "worth" through dependency information, thereby making a correct prediction. The ASGCN module based on the attention mechanism can understand the semantics well and give a correct prediction. Triple-GCN contains two modules, SynGCN and ASGCN, so it can correctly predict the sentiment polarity of the sample 1.

From the prediction results of the sample 2, we can observe that ASGCN can effectively reduce the negative impact of dependency parsing. The dependency tree of the sample 2 and the attention weight in ASGCN are shown in Figure 6. In the dependency tree, the aspect word "settings" is directly connected to the positive word "convenient", leading SynGCN make a incorrect prediction (Positive). However, ASGCN can capture deep semantic information, making "not" have higher weight, consequently producing a correct prediction (Negative). After integrating the feature information of SynGCN and ASGCN, Triple-GCN can make correct predictions.

In the sample 3, unlike sample 2, the word "settings" in the dependency tree is not linked to the word "fine" which indicates a positive implication, which enables the syntax-based SynGCN model to provide the accurate answer. In the ASGCN's attention weight matrix, "fine" possesses heightened weight, leading to a prediction error on ASGCN's part. After combining the feature information of SynGCN and ASGCN, Triple-GCN persists in rendering the correct prediction. These three sentences further prove the effectiveness of Triple-GCN.

4.7 Attention Visualization

With the effectiveness of SynGCN and ASGCN in capturing semantic relevance, respectively. We aim to visualize the attention score matrix. Figure 7 shows the visualization of attention distribution for words. Observably, when using only the syntax or semantic convolution module, the model erroneously directs the highest attention to "wonderful".

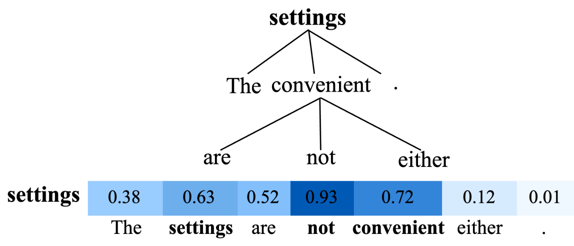

Figure 6. Dependency trees and attention weights

Treating "wonderful" as a viewpoint word of aspect "dinner" ultimately leads to misjudgment of the sentence. However, the proposal of Triple-GCN (line 3) can adaptively reduce attention to the irrelevant word "wonderful", adaptively increase the score of "dinner", and obtain the most relevant words from syntax, semantics and their combinations information, verifying the effectiveness of the Triple-GCN model.

Misinterpreting "wonderful" as a viewpoint word relative to the aspect "dinner" ultimately results in an incorrect judgment of the sentence. Conversely, the implementation of Triple-GCN (line 3) can adeptly lessen the attention on the irrelevant word "wonderful", increment the score of "dinner" adaptively, and secure the most relevant words from syntax, semantics, and their combined information. This evidences the efficiency of our Triple-GCN model.

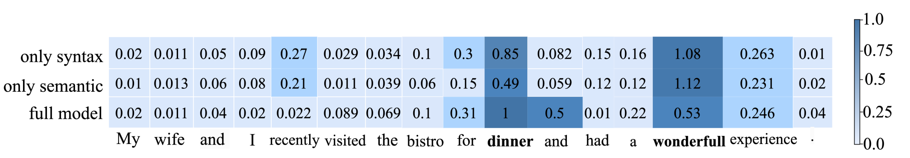

Figure 7. Attention visualization for sentence 1

Again, we take the sentence "The environment is great but the service attitude is poor." in Figure 8 as a sample to illustrate attention visualization. As shown in Figure 8, the darker the color of the area block, the greater the attention weight and the higher the attention. Figure 8 also presents the attention weights of the two aspect terms and their corresponding attention weights within the contexts. When "environment" is used as an aspect term, the model allocates higher attention to the descriptive word "great" within the context. Concurrently, the model's attention is predominantly focused on the relevant descriptions of "environment", while rarely attending to "service attitude". Similarly, when "service attitude" is the aspect term, the model directs greater attention to the relevant descriptions of "service attitude", demonstrating the attention mechanism's ability to focus on words associated with the aspect terms. In particular, when an aspect term is composed of multiple words, the model calculates the weight of each word within the aspect term. For example, in the case of "service attitude", "attitude" is the primary word, while "service" is used to modify it. As such, "attitude" should hold greater importance in expressing the overall aspect term. It can also be observed from Figure 8 that the model indeed allocates greater attention to "attitude", which demonstrates the model's ability to identify the more salient word information within multi-word aspect terms.

Therefore, the interactive attention mechanism not only allows the model to focus on the interaction between context and aspect terms, but also enables it to attend to the meaning of the context and aspect terms themselves. This provides richer information for recognizing the sentiment polarity of aspect terms.

Figure 8. Attention visualization for sentence 2

4.8 Hyper-Parameter

The results of the study on the impact of the hyper-parameter m value and the number of attention heads M of the semantic graph convolution module are shown below.

- **The influence of the m value:** According to the results in Figure 9, on the three datasets, the model performance is best when the value of m is 2, 2, and 4, respectively. As the value of m increases, the accuracy of the model on the three datasets decreases. In summary, a larger value of m will introduce more irrelevant noise, thereby interfering with the model's judgment of sentiment polarity.
- **The influence of the number of attention heads M**: Taking the Twitter dataset as an example, Figure 11 shows the impact of the number of heads M of the multi-head attention mechanism in the model on performance. Intuitively, when the number of heads is 3, the model performance is best, and too small or too large a number of heads will affect the model effect. Therefore, m is set to 2 or 4 in the Triple-GCN model, and the number of attention heads M is set to 3 to achieve the best performance.
- **The influence of the hyper-parameter $\boldsymbol{\alpha}$**: In order to optimize the hyper-parameter α, an experimental design was utilized. This exercise revolved around calibrating α across the 0.0 to 1.0 range and assessing how these alterations impacted model effectiveness. Upon application to the Twitter, Laptop14, and

Restaurant14 datasets, the ideal α varied, attributed to each corpus' unique data attributes. For the Twitter dataset, where tweets are typically brief and semantic information is predominant, a relatively high α of 0.8 achieved optimal precision. In contrast, the Laptop14 and Restaurant14 datasets, characterized by detailed and complex review inputs, revealed a dependency on type information. Consequently, a more moderate α value of 0.6 yielded the greatest precision. The integral role of adjusting α to enhance model efficiency across varied datasets was validated through this exercise.

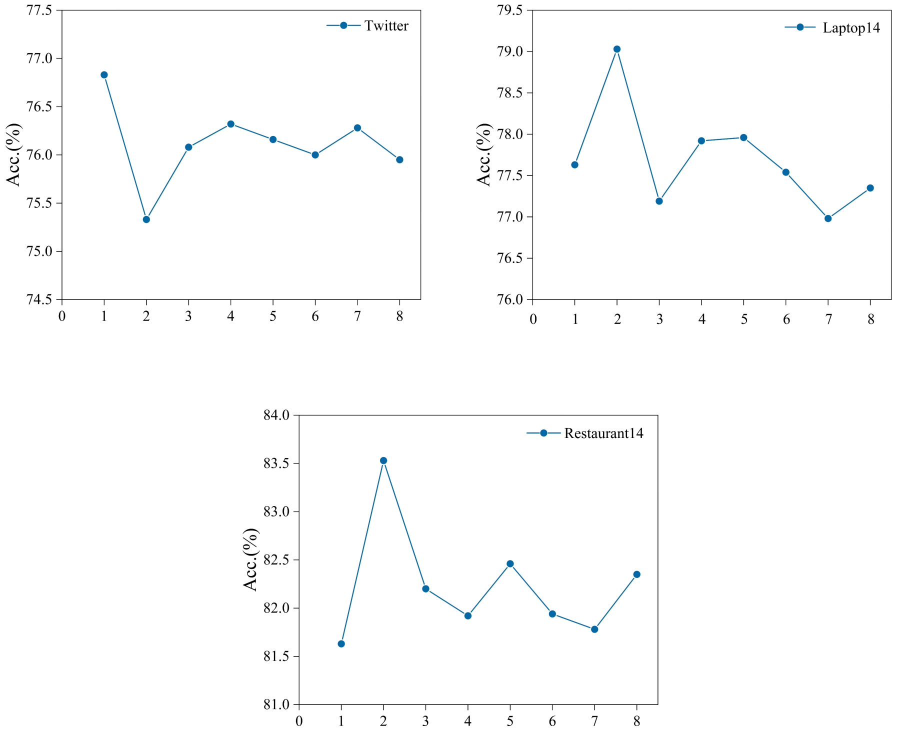

Figure 9. The influence of the m value

4.9 Number of Model Layers

To thoroughly explore the impact of layer count on the performance of the Triple-GCN model, we conducted a series of experiments across three distinct datasets: Twitter, Laptop14, and Restaurant14. These experiments were specifically designed

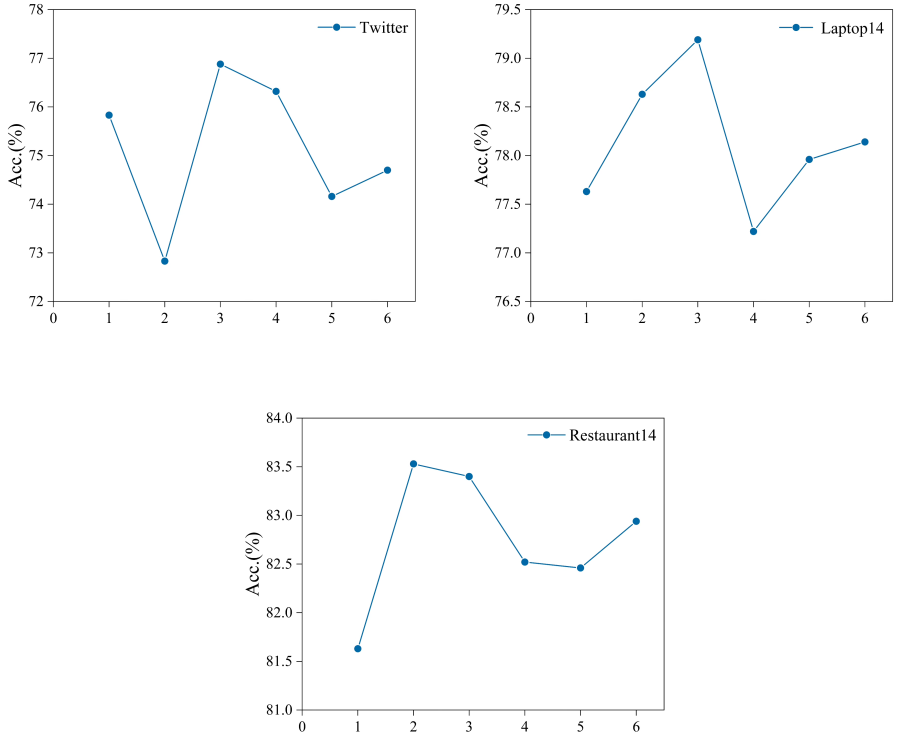

Figure 10. The influence of the number of attention heads M

to monitor how the model's performance varied as the number of layers was systematically increased from 1 to 5. The findings from these experiments are visually represented in Figure 12.

The Laptop14 and Restaurant14 datasets, both of which possess standard syntactic structures, exhibit similar performance trends when the Triple-GCN model is applied. As indicated in Figure 12, the model achieves its peak performance on these datasets when the number of layers is set to $N_1 = 2$. This suggests that for datasets with more conventional and structured syntax, a two-layer configuration of the Triple-GCN model is sufficient to capture the essential syntactic and semantic information, providing an optimal balance between model complexity and performance.

Conversely, the performance trend observed on the Twitter dataset, characterized by its non-standard syntactic structure, differs noticeably from that of Laptop14 and Restaurant14. In this case, the Triple-GCN model attains its highest performance when the number of layers is set to $N_1 = 3$. This indicates that the additional

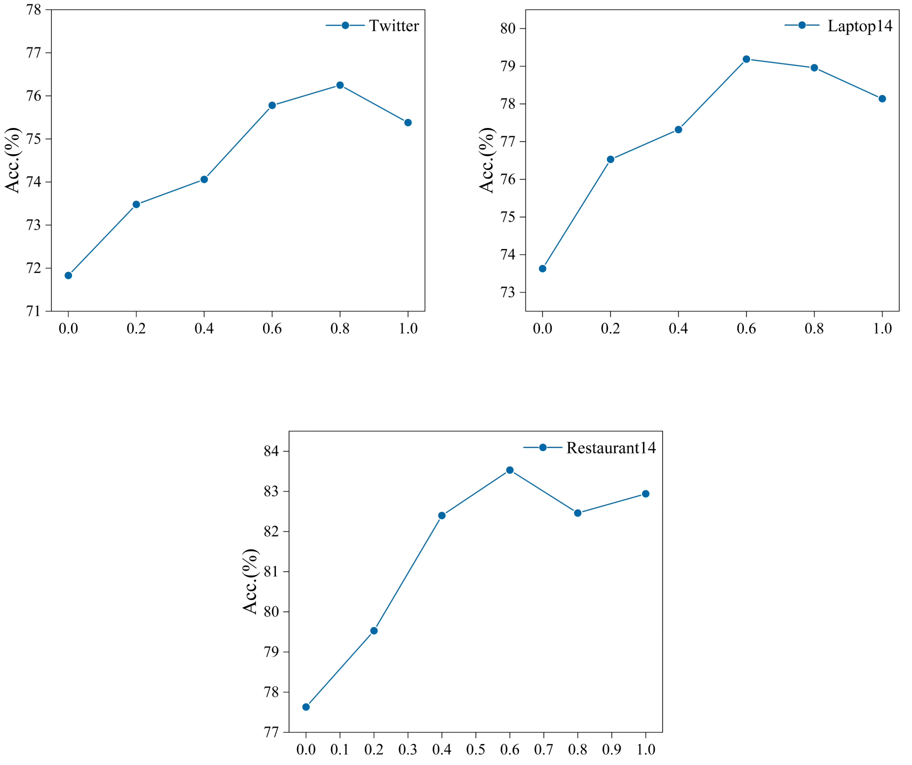

Figure 11. The influence of the hyper-parameter α

layer helps the model better adapt to the informal and often fragmented language typical of social media texts found on Twitter, where syntactic rules are less rigid and more variable.

Overall, these results underscore the necessity of tailoring the number of layers in the Triple-GCN model to the specific syntactic characteristics of the dataset being analyzed. For datasets with well-structured and conventional syntax, fewer layers may suffice. In contrast, datasets with less structured and more complex syntax, such as those from social media, may benefit from additional layers to enhance performance.

5 CONCLUSION

In response to the under-utilization of syntactic structures and the overlooked link between syntax and semantics, this study introduces an innovative aspect-level sen-

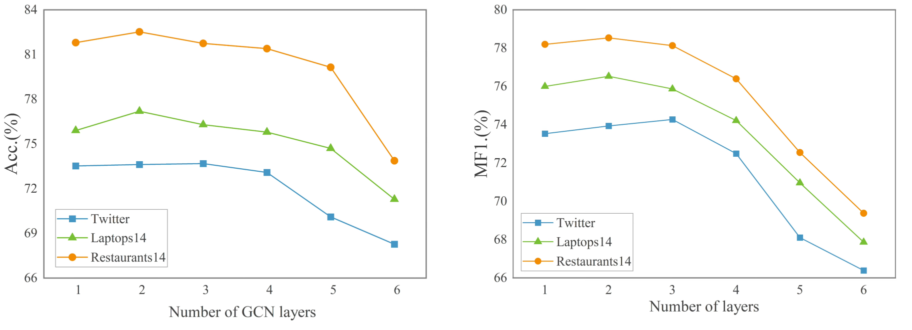

Figure 12. The influence of GCN layers

timent analysis model called Triple-GCN. This model leverages a multi-channel graph convolutional network architecture that includes three carefully designed modules: Syntactic Graph Convolution (SynGCN), Semantic Graph Convolution based on Attention Mechanism (ASGCN), and Shared Enhanced Graph Convolution (SEGCN). Each module contributes uniquely to the model's effectiveness. SynGCN focuses on mining syntactic information, especially through position weight encoding that improves the understanding of sentiment dependency relations. ASGCN emphasizes semantic information linked to key nodes, while SEGCN bridges the gap between syntax and semantics, ensuring optimal utilization of both. Experimental verification on three widely used benchmark datasets has demonstrated that Triple-GCN effectively integrates relation-type information to connect syntax and semantics, resulting in superior performance in aspect-level sentiment analysis. Compared to similar studies, such as DualGCN and MGAN, Triple-GCN shows a more balanced integration of syntactic and semantic features, leading to enhanced accuracy and generalization capabilities. Despite these advances, current research, including this study, still faces challenges in fully capturing the complexities of sentiment information, especially in nuanced or context-dependent scenarios. The insights gained from this study can guide future research, particularly in refining sentiment analysis models through the integration of sentiment knowledge graphs, lexicon knowledge, and common-sense knowledge. As the field progresses, the fusion of syntax and semantics is likely to become even more crucial, leading to models with greater expressiveness and applicability in areas such as customer feedback analysis, brand reputation management, and social media monitoring.

REFERENCES

[1] Liu, B.: Sentiment Analysis and Opinion Mining. Springer Nature, 2022, https://www.cs.uic.edu/~liub/FBS/liub-SA-and-OM-book.pdf.

[2] Jiang, L.—Yu, M.—Zhou, M.—Liu, X.—Zhao, T.: Target-Dependent Twitter Sentiment Classification. In: Lin, D., Matsumoto, Y., Mihalcea, R. (Eds.): Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL 2011). 2011, pp. 151–160, https://aclanthology.org/P11-1016.

[3] Pontiki, M.—Galanis, D.—Papageorgiou, H.—Androutsopoulos, I.—Manandhar, S. et al.: SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In: Bethard, S., Carpuat, M., Cer, D., Jurgens, D., Nakov, P., Zesch, T. (Eds.): Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016). 2016, pp. 19–30, doi: 10.18653/v1/S16-1002.

[4] Hu, M.—Liu, B.: Mining and Summarizing Customer Reviews. Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '04), 2004, pp. 168–177, doi: 10.1145/1014052.1014073.

[5] Titov, I.—McDonald, R.: Modeling Online Reviews with Multi-Grain Topic Models. Proceedings of the 17th International Conference on World Wide Web (WWW '08), 2008, pp. 111–120, doi: 10.1145/1367497.1367513.

[6] Kiritchenko, S.—Zhu, X.—Cherry, C.—Mohammad, S.: NRC-Canada-2014: Detecting Aspects and Sentiment in Customer Reviews. Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), 2014, pp. 437–442, doi: 10.3115/v1/S14-2076.

[7] Chen, P.—Sun, Z.—Bing, L.—Yang, W.: Recurrent Attention Network on Memory for Aspect Sentiment Analysis. In: Palmer, M., Hwa, R., Riedel, S. (Eds.): Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017). 2017, pp. 452–461, doi: 10.18653/v1/D17-1047.

[8] Tang, D.—Qin, B.—Feng, X.—Liu, T.: Effective LSTMs for Target-Dependent Sentiment Classification. In: Matsumoto, Y., Prasad, R. (Eds.): Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. 2016, pp. 3298–3307, https://aclanthology.org/C16-1311/.

[9] Tan, X.—Cai, Y.—Zhu, C.: Recognizing Conflict Opinions in Aspect-Level Sentiment Classification with Dual Attention Networks. In: Inui, K., Jiang, J., Ng, V., Wan, X. (Eds.): Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019). 2019, pp. 3426–3431, doi: 10.18653/v1/D19-1342.

[10] Gu, S.—Zhang, L.—Hou, Y.—Song, Y.: A Position-Aware Bidirectional Attention Network for Aspect-Level Sentiment Analysis. In: Bender, E. M., Derczynski, L., Isabelle, P. (Eds.): Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018). 2018, pp. 774–784, https://aclanthology.org/C18-1066/.

[11] Wang, Y.—Huang, M.—Zhu, X.—Zhao, L.: Attention-Based LSTM for Aspect-Level Sentiment Classification. In: Su, J., Duh, K., Carreras, X. (Eds.): Proceed-

ings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016). 2016, pp. 606–615, doi: 10.18653/v1/D16-1058.

[12] Zhao, P.—Hou, L.—Wu, O.: Modeling Sentiment Dependencies with Graph Convolutional Networks for Aspect-Level Sentiment Classification. Knowledge-Based Systems, Vol. 193, 2020, Art. No. 105443, doi: 10.1016/j.knosys.2019.105443.

[13] Kipf, T. N.—Welling, M.: Semi-Supervised Classification with Graph Convolutional Networks. International Conference on Learning Representations (ICLR 2017), 2017, doi: 10.48550/arXiv.1609.02907.

[14] Veličković, P.—Cucurull, G.—Casanova, A.—Romero, A.—Liò, P.—Bengio, Y.: Graph Attention Networks. International Conference on Learning Representations (ICLR 2018), 2018, doi: 10.48550/arXiv.1710.10903.

[15] Chang, M.—Yang, M.—Jiang, Q.—Xu, R.: Counterfactual-Enhanced Information Bottleneck for Aspect-Based Sentiment Analysis. Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 38, 2024, No. 16, pp. 17736–17744, doi: 10.1609/aaai.v38i16.29726.

[16] Pang, S.—Xue, Y.—Yan, Z.—Huang, W.—Feng, J.: Dynamic and Multi-Channel Graph Convolutional Networks for Aspect-Based Sentiment Analysis. In: Zong, C., Xia, F., Li, W., Navigli, R. (Eds.): Findings of the Association for Computational Linguistics (ACL-IJCNLP 2021). 2021, pp. 2627–2636, doi: 10.18653/v1/2021.findings-acl.232.

[17] Li, R.—Chen, H.—Feng, F.—Ma, Z.—Wang, X.—Hovy, E.: Dual Graph Convolutional Networks for Aspect-Based Sentiment Analysis. In: Zong, C., Xia, F., Li, W., Navigli, R. (Eds.): Proceedings of the 59$^{\text{th}}$ Annual Meeting of the Association for Computational Linguistics and the 11$^{\text{th}}$ International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (ACL-IJCNLP 2021). 2021, pp. 6319–6329, doi: 10.18653/v1/2021.acl-long.494.

[18] Feng, J.—Cai, S.—Li, K.—Chen, Y.—Cai, Q.—Zhao, H.: Fusing Syntax and Semantics-Based Graph Convolutional Network for Aspect-Based Sentiment Analysis. International Journal of Data Warehousing and Mining (IJDWM), Vol. 19, 2023, No. 1, doi: 10.4018/IJDWM.319803.

[19] Pylkkänen, L.: The Neural Basis of Combinatory Syntax and Semantics. Science, Vol. 366, 2019, No. 6461, pp. 62–66, doi: 10.1126/science.aax0050.

[20] Pylkkänen, L.: Neural Basis of Basic Composition: What We Have Learned from the Red-Boat Studies and Their Extensions. Philosophical Transactions of the Royal Society B, Vol. 375, 2020, No. 1791, doi: 10.1098/rstb.2019.0299.

[21] Ma, D.—Li, S.—Zhang, X.—Wang, H.: Interactive Attention Networks for Aspect-Level Sentiment Classification. Proceedings of the 26$^{\text{th}}$ International Joint Conference on Artificial Intelligence (IJCAI '17), 2017, pp. 4068–4074, doi: 10.48550/arXiv.1709.00893.

[22] Fan, F.—Feng, Y.—Zhao, D.: Multi-Grained Attention Network for Aspect-Level Sentiment Classification. In: Riloff, E., Chiang, D., Hockenmaier, J., Tsujii, J. (Eds.): Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018). 2018, pp. 3433–3442, doi: 10.18653/v1/D18-1380.

[23] Zhang, C.—Li, Q.—Song, D.: Aspect-Based Sentiment Classification with

Aspect-Specific Graph Convolutional Networks. In: Inui, K., Jiang, J., Ng, V., Wan, X. (Eds.): Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019). 2019, pp. 4568–4578, doi: 10.18653/v1/D19-1464.

[24] Huang, B.—Carley, K.: Syntax-Aware Aspect Level Sentiment Classification with Graph Attention Networks. In: Inui, K., Jiang, J., Ng, V., Wan, X. (Eds.): Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019). 2019, pp. 5469–5477, doi: 10.18653/v1/D19-1549.

[25] Wang, K.—Shen, W.—Yang, Y.—Quan, X.—Wang, R.: Relational Graph Attention Network for Aspect-Based Sentiment Analysis. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (Eds.): Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020). 2020, pp. 3229–3238, doi: 10.18653/v1/2020.acl-main.295.

[26] Sun, K.—Zhang, R.—Mensah, S.—Mao, Y.—Liu, X.: Aspect-Level Sentiment Analysis via Convolution over Dependency Tree. In: Inui, K., Jiang, J., Ng, V., Wan, X. (Eds.): Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019). 2019, pp. 5679–5688, doi: 10.18653/v1/D19-1569.

[27] 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: Scott, D., Bel, N., Zong, C. (Eds.): Proceedings of the 28th International Conference on Computational Linguistics (COLING 2020). 2020, pp. 150–161, doi: 10.18653/v1/2020.coling-main.13.

[28] Zhang, M.—Qian, T.: Convolution over Hierarchical Syntactic and Lexical Graphs for Aspect Level Sentiment Analysis. In: Webber, B., Cohn, T., He, Y., Liu, Y. (Eds.): Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020). 2020, pp. 3540–3549, doi: 10.18653/v1/2020.emnlp-main.286.

[29] Devlin, J.—Chang, M. W.—Lee, K.—Toutanova, K.: BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In: Burstein, J., Doran, C., Solorio, T. (Eds.): Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (NAACL 2019). 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.

[30] Dong, L.—Furu, W.—Tan, C.—Tang, D.—Zhou, M.—Xu, K.: Adaptive Recursive Neural Network for Target-Dependent Twitter Sentiment Classification. In: Toutanova, K., Wu, H. (Eds.): Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014). 2014, pp. 49–54, doi: 10.3115/v1/P14-2009.

[31] Pontiki, M.—Galanis, D.—Pavlopoulos, J.—Papageorgiou, H.—Androutsopoulos, I.—Manandhar, S.: SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In: Nakov, P., Zesch, T. (Eds.): Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). 2014, pp. 27–35,

doi: 10.3115/v1/S14-2004.

[32] Manning, C. D.—Surdeanu, M.—Bauer, J.—Finkel, J.—Bethard, S.—McClosky, D.: The Stanford CoreNLP Natural Language Processing Toolkit. In: Bontcheva, K., Zhu, J. (Eds.): Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations (ACL 2014). 2014, pp. 55–60, doi: 10.3115/v1/P14-5010.

[33] Pennington, J.—Socher, R.—Manning, C.: GloVe: Global Vectors for Word Representation. In: Moschitti, A., Pang, B., Daelemans, W. (Eds.): Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014). 2014, pp. 1532–1543, doi: 10.3115/v1/D14-1162.

[34] Han, H.—Wu, Y.—Qin, X.: An Interactive Graph Attention Networks Model for Aspect-Level Sentiment Analysis. Journal of Electronics & Information Technology, Vol. 43, 2021, No. 11, pp. 3282–3290, doi: 10.11999/JEIT210036.

[35] Tang, H.—Ji, D.—Li, C.—Zhou, Q.: Dependency Graph Enhanced Dual-Transformer Structure for Aspect-Based Sentiment Classification. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (Eds.): Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020). 2020, pp. 6578–6588, doi: 10.18653/v1/2020.acl-main.588.

Huanling Tang is currently a Professor with the School of Computer Science and Technology at Shandong Technology and Business University, Yantai, China. Her current research interests include deep learning, machine learning, and natural language processing.

Xueyuan Sun is a Master student in the School of Information and Electronic Engineering at Shandong Technology and Business University, Yantai, China. Her current research interests include aspect-based sentiment analysis and deep learning.

Quansheng Dou is currently a Professor with the School of Computer Science and Technology at Shandong Technology and Business University, Yantai, China. His current research interests include machine learning, deep learning, and natural language processing.

Mingyu Lu is currently the Deputy Dean of scientific research at the School of Information Science and Technology, Dalian Maritime University. His current research interests include machine learning, big data, and deep learning.