

Article

SSEMGAT: Syntactic and Semantic Enhanced Multi-Layer Graph Attention Network for Aspect-Level Sentiment Analysis

Xiangzhe Xin ¹, Aishan Wumaier ² , Zaokere Kadeer ^{2,*} and Jiangtao He ¹¹ College of Information Science and Engineering, Xinjiang University, Urumqi 830046, China² Xinjiang Laboratory of Multi-Language Information Technology, College of Information Science and Engineering, Xinjiang University, Urumqi 830046, China* Correspondence: zuhra@xju.edu.cn

Abstract: Aspect-level sentiment analysis aims to identify the sentiment polarity of specific aspects appearing in a given sentence or review. The model based on graph structure uses a dependency tree to link the aspect word with its corresponding opinion word and achieves significant results. However, for some sentences with ambiguous syntactic structure, it is difficult for the dependency tree to accurately parse the dependencies, which introduces noise and degrades the performance of the model. Based on this, we propose a syntactic and semantic enhanced multi-layer graph attention network (SSEMGAT), which introduces constituent trees in syntactic features to compensate for dependent trees at the clause level, exploiting aspect-aware attention in semantic features to assign the attention weight of specific aspects between contexts. The enhanced syntactic and semantic features are then used to classify specific aspects of sentiment through a multi-layer graph attention network. Accuracy and Macro-F1 are used as evaluation indexes in the SemEval-2014 Task 4 Restaurant and Laptop dataset and the Twitter dataset to compare the proposed model with the baseline model and the latest model, achieving competitive results.

Keywords: aspect-level sentiment analysis; graph attention network; feature extract



Citation: Xin, X.; Wumaier, A.; Kadeer, Z.; He, J. SSEMGAT: Syntactic and Semantic Enhanced Multi-Layer Graph Attention Network for Aspect-Level Sentiment Analysis. *Appl. Sci.* **2023**, *13*, 5085. <https://doi.org/10.3390/app13085085>

Academic Editor: Christos Bouras

Received: 14 March 2023

Revised: 17 April 2023

Accepted: 17 April 2023

Published: 19 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The rapid development of the Internet has changed people's way of life. For example, information is exchanged and shared through online service platforms, which generates a large number of comment information. These comments not only contain users' views and attitudes towards news events, which can help the government and other agencies monitor public opinion, but also contain preferences for products, which can help commercial companies quickly complete product analysis and make improvements. These comment data have great social and commercial value. It is of great significance to use sentiment analysis technology to study these comments. Aspect-level sentiment analysis is a subtask in sentiment analysis. It is a fine-grained sentiment analysis task, aiming at judging the sentiment tendency of different aspects of entities in comments. Recently, the syntax-based model has used the dependency tree to extract syntactic information and apply it to the aspect-level sentiment analysis task, which has achieved remarkable results. Dependency trees capture dependencies between aspect words and their corresponding opinion words, which can solve the problem of long-distance dependence [1]. Therefore, they are often used to extract syntactic information. Due to the arbitrary expression of online comments, there is no obvious syntactic structure, which leads to the introduction of noise (irrelevant dependency relation) in the parsing of dependency-tree-based methods, reducing the ability of a dependency tree to capture the sentiment-aware context [2].

Based on the above observations, we propose a syntactic and semantic enhanced multi-layer graph attention network (SSEMGAT). The dependency tree is used to represent the dependency between words at the word level; the constituent tree is introduced to

obtain syntactic information from a higher-level perspective. The attention mechanism is easily disturbed by other aspect words. It uses aspect-aware attention to redistribute the attention weight between specific aspect words and context. Then, the extracted syntactic and semantic features are fed into the multi-layer graph attention module for specific aspects of sentiment classification.

The main contributions of this paper are as follows:

- (1) For the aspect-level sentiment analysis task, we propose a syntactic and semantic enhanced multi-layer graph attention network to extract features from syntactic and semantic perspectives and use pre-training knowledge to integrate syntactic and semantic features extracted to infer specific aspects of sentiment polarity.
- (2) We introduce a constituent tree to make up for the defect in the dependency tree and combine different levels of syntactic information to align the position of the aspect word and its corresponding opinion word. At the same time, aspect-aware attention and multi-headed attention are used to construct local attention and global attention, respectively, to link sentiment information between specific aspects and contexts.
- (3) Experimental results on three benchmark datasets show that the performance of the SSEMGAT model exceeds the baseline model and some recent models. Our model incorporates syntactic and semantic feature information well, which indicates that our work is effective.

The following sections of this paper are arranged as follows: In Section 2, we introduce the relevant work of aspect-level sentiment analysis, which is mainly divided into three categories: attention-based approach, syntax-based approach, and pre-training-based approach. In Section 3, we describe the proposed model in detail. In Section 4, we test our proposed model on the public benchmark datasets and analyze it separately. Finally, in Section 5, we summarize the whole paper and look forward to future work.

2. Related Work

Sentiment analysis (SA) is an important research direction in opinion mining. It is the process of using natural language processing technology (NLP) to analyze and summarize text content containing sentiment. Sentiment analysis is divided into sentence-level [3,4], chapter-level [5,6], and aspect-level analysis. The sentence level aims at comment text, which needs to judge its whole sentiment tendency and provide corresponding sentiment values, generally including positive, neutral, and negative. Chapter level refers to a document, which judges the overall sentiment tendency and provides the same sentiment value as the sentence level. Both methods judge the whole and generally only provide sentiment value, which belongs to coarse-grained sentiment. Aspect level aims at the multiple aspects of the entity contained in the review text; each aspect can be composed of different sentiment values, and different aspects can have different sentiment values, even conflict, while the sentence level and chapter level only have one direction of sentiment. Existing studies on aspect-level sentiment analysis can be broadly split into three categories:

- (1) Attention-based methods: The attention mechanism models the dependency relationship between an aspect term and its corresponding opinion words. However, there may be several different aspect terms in a sentence. There have been studies to judge the sentiment of a particular aspect. Wang et al. [7] captured the importance of different contextual information to a given aspect word through the attention mechanism, and the attention mechanism and LSTM are combined to model the semantics of sentences and solve the problem of aspect-level sentiment analysis. Ma et al. [8] proposed an interactive attention network (IAN), which uses the attention mechanism to link the target and context for multi-level semantic classification. Chen et al. [9] used multiple attention mechanisms to capture connections between long-distance sentiment features, with strong robustness to irrelevant information. Huang et al. [10] introduced an attention-over-attention (AOA) module to capture the connection between aspects and context words. Fan et al. [11] proposed a multi-grained attention network (MGAN) to combine coarse-grained and fine-grained attention to capture

- the interaction of aspect and context at the word level. The attention-based approach achieves attractive results. However, due to its defect, the attention mechanism is easily affected by the noise in the sentence, thus misjudging the sentiment polarity.
- (2) **Syntax-based methods:** Some work explicitly uses dependency trees of a sentence to extract syntactic information. Zhang et al. [12] first proposed building a graph convolutional neural network on a dependency tree to learn the dependencies between nodes. Sun et al. [13] utilized the representation of sentence features learned from the bidirectional LSTM and enhanced embedding with the graph convolutional network. Zhang et al. [14] constructed a hierarchical syntactic graph and lexical graph via convolution on GNN embedding and BiLSTM embedding, respectively, and a bi-level interactive network was designed to learn information interaction. Chen et al. [15] combined information from the latent graph and the dependency graph via a gated attention mechanism. For the situation where the current node of the dependency tree pays average attention to adjacent nodes, Wang et al. [16] constructed an aspect-oriented dependency tree structure (R-GAT) by extending the graph attention network to encode graphs with labeled edges. Most syntax-based models only make use of dependency, without considering the type of dependency. Tian et al. [17] proposed T-GCN, which uses an attention mechanism to distinguish different edges in a graph and uses attention layer ensemble to comprehensively learn different layers of T-GCN. The use of syntactic knowledge only cannot obtain the best results, and some researchers have studied the use of other knowledge. Li et al. [18] proposed a dual graph convolutional neural network (DualGCN) to construct syntactic graphs and semantic graphs from the perspective of syntactic structure and semantic correlation, respectively. Zhang et al. [2] combined the attention matrix constructed by the attention mechanism and syntactic mask matrix to accomplish the interaction of syntactic structure and semantic information. Wu et al. [19] used a dependency tree and phrase tree to construct a phrase dependency graph and used the PD-RGAT model on it for the ABSA task. Compared with the attention-based model, the performance of the syntax-based method was greatly improved, but some shortcomings cannot be ignored. Since dependency trees have different syntactic sensitivities, the noise introduced to sentences without obvious syntactic structure will make it difficult for dependency trees to accurately capture sentiment-aspect context [17], and GCN cannot perfectly integrate topological structure and node features [20]. These problems limit the further development of graph neural networks.
 - (3) **Pre-trained-based methods:** Devlin et al. [21] used the left and right context to pre-train the depth bidirectional representation, requiring only one additional output layer to fine-tune the pre-trained BERT representation, achieving state-of-the-art results for a variety of tasks without basic task-specific architecture modifications. Xu et al. [22] proposed training on large-scale general domain data and fine-tuning on a small amount of downstream data, which provides a solution for the study of small sample data. Song et al. [23] designed an attentional encoder to generate hidden representations, and the BERT-SPC model is designed as a comparison model for sentence pair classification tasks. There are also some studies using a combination of pre-training and GCN. Jawahar et al. [24] found that BERT could capture a rich hierarchy of language information, with phrase features at the bottom, syntactic features in the middle, and semantic features at the top. Xiao et al. [25] integrated syntactic sequence information from BERT and knowledge from dependency trees to enhance graph convolutional neural networks for better coding dependency graphs. Tang et al. [26] regarded GCN as a special form of transformer and studied the representation between GCN and a transformer interactively.

3. Methodology

In this section, we introduce the syntactic and semantic enhanced multi-layer graph attention model, that is, SSEMGAT. The overall structure of the model is shown in Figure 1.

It is mainly divided into four parts: input layer, extraction layer, MGAT module, and fusion layer. Next, we will describe each module in the model in detail.

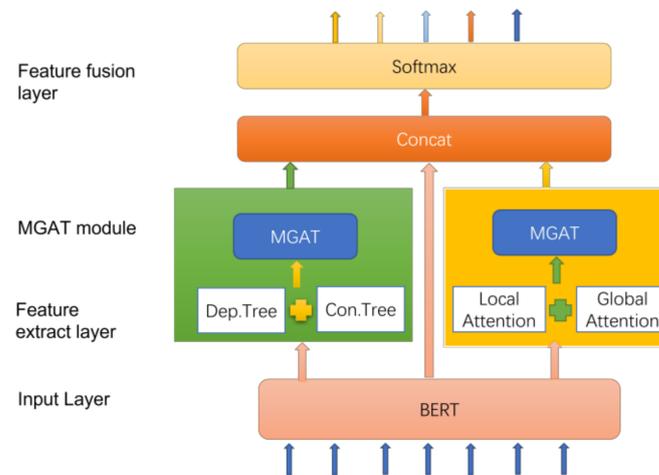


Figure 1. The framework of the proposed SSEMGAT model.

3.1. Input Layer

Given a sentence of n words $s = \{\omega_1, \omega_2, \dots, a_1, a_2, \dots, a_m, \dots, \omega_n\}$, where $\{a_1, a_2, \dots, a_m\}$ is aspect term, since BERT has a powerful representation learning capacity, we utilize BERT as a sentence encoder to generate contextual representations. To accommodate the input format of the BERT model, given target aspect, we follow BERT-SPC [23] to construct a BERT-based sequence: [CLS] + {sentence} + [SEP] + aspect + [SEP]. However, there may be multiple aspects in a sentence, so we use the form of [CLS] + {sentence} + [SEP] + aspect + [SEP] + aspect + [SEP] to construct the pattern sequence. Then, the output representation H is obtained by BERT,

$$H = \{h_1^t, \dots, h_n^t\} \quad (1)$$

3.2. Extraction Layer

The existing models based on graph structure often use the dependency tree to extract syntactic information, the attention mechanism to extract semantic information, and use GCN to construct syntactic graphs and semantic graphs; the above graphs are interactively learned, and good results are achieved.

3.2.1. Syntactic Feature Extraction

Generally, a dependency tree (Dep.Tree) can capture dependencies between aspect terms and their corresponding opinion words, maintaining valid in the long-distance dependency problem. Therefore, dependency trees are often used to extract syntactic information from sentences. However, not all information on the dependency tree is beneficial to our task, and introducing noise (unrelated relations of dependencies) makes it difficult for each aspect word to accurately capture the corresponding contextual sentiment information. For example, the dependency tree parsing of sentences is shown in Figure 2, and the “conj” relation between “delicious” and “terrible” is invalid for our task, but the aspect term “taste” may be associated with the opinion word “terrible”, reducing the ability to accurately capture “delicious” in the opinion words.

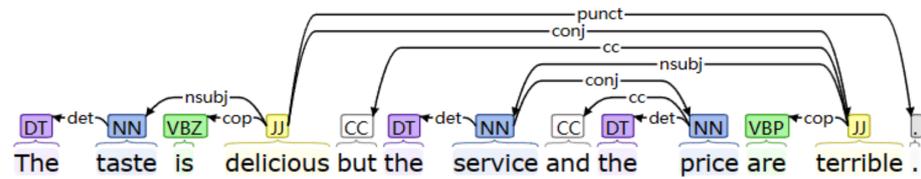


Figure 2. The result of dependency tree parses.

Moreover, the dependency tree reveals relations between words, the relationship between clauses and between aspects that is difficult to capture. Based on this, we use constituent trees, which mainly consist of phrase segmentation and hierarchical structures that help to correctly align aspect words with their corresponding opinion words of sentiment information. Phrase segmentation can easily divide a sentence into multiple clauses and refine the syntactic position of each word in the sentence. The structured hierarchy can distinguish different relationships between aspect words to infer different aspects of sentiment information from a clause-level perspective. For example, the result of parsing the constituent tree of sentences is shown in Figure 3. The whole sentence is divided into four parts: clause “The taste is delicious”, phrase segmentation term “but”, clause “the service and price are terrible”, and “.”. In hierarchical structure, according to the phrase segmentation term “and”, we can find that the aspect words “service” and “price” have the same sentiment polarity, while according to the phrase segmentation term “but”, it is concluded that it has the opposite sentiment polarity towards the aspect word “taste” and the aspect words of other clauses.

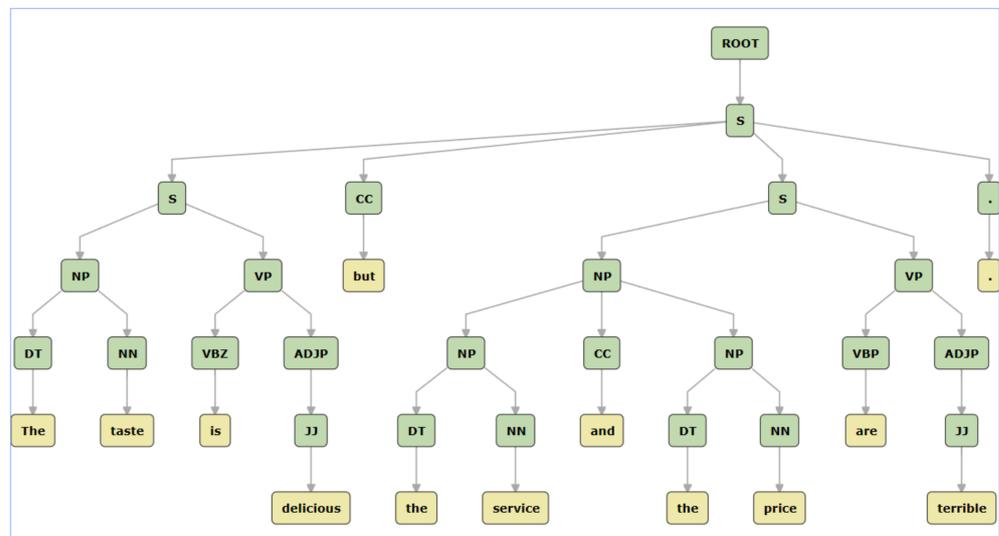


Figure 3. The result of constituent tree parses.

Integrating information from different structural levels can obtain more accurate syntactic information. Therefore, we construct the dependency adjacency matrix DA at the word level and the constituent adjacency matrix CA from the clause level, which is constructed as follows:

- (1) Matrix DA : Using the dependency tree as an undirected graph, if there is a connection between the words w_i and w_j ,

$$DA_{i,j} = \begin{cases} 1, & \text{if } w_i, w_j \text{ link directly in } Dep.Tree \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

- (2) Matrix CA: The constituent tree has a hierarchical structure, and in each layer, if words w_i and w_j belong to the same clause phrase,

$$CA_{i,j}^l = \begin{cases} 1, & \text{if } w_i, w_j \text{ in same phrase of } \{ph_u^l\} \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

Then, the CA and DA matrices are combined via position-wise addition as the extracted syntactic feature matrix A_{syn} :

$$A_{syn} = CA + DA \quad (4)$$

3.2.2. Semantic Feature Extraction

Attention mechanism is a common way to capture the interactions between the aspect and context words. However, the attention mechanism is easily disturbed by noise (other irrelevant aspects of words), and as clues, misjudge the sentiment polarity of the related aspects. Therefore, we use aspect-aware attention to learn local semantic information for a specific aspect, while using self-attention to learn global semantic information for sentences. After that, we fuse local attention with global attention to learn semantic correlation.

- (1) Local attention: To enhance the attention of specific aspects to local contextual sentiment information, we use aspect-aware attention to prevent disturbance with other aspects of word information. The aspect-aware attention mechanism utilizes the aspect term as query conditions to calculate the attention feature information of related aspects,

$$A_{local}^i = \tanh\left(H_a W^a * (KW^K)^T + b\right) \quad (5)$$

where K is equal to the output H of the input layer, and W^a and W^K are learnable weights. We perform mean pool operation on output H and copy the processed output n times as H_a .

- (2) Global attention: The attention mechanism captures the semantic correlation between any two words in a sentence. This is useful for grasping all of the semantic information in a sentence. Therefore, we use the multi-head attention mechanism [27] to construct the global semantic score matrix A_{global}^i of the sentence. The calculation process is as follows,

$$A_{global}^i = \text{soft}\left(\frac{QW^Q * (KW^K)^T}{\sqrt{d}}\right) \quad (6)$$

where W^Q and W^K are learnable weights

Then, we combine the local attention score with the global score to obtain semantic matrix A_{sem} :

$$A_{sem} = A_{global} + A_{local} \quad (7)$$

3.3. Multi-Layer Graph Attention Module (MGAT)

To utilize rich hierarchical syntactic information, we use the MGAT block stacked by several designed graph attention layers [28]. GAT is a new graph neural network architecture, including an attention mechanism, which enables one to assign different attention weights to the information provided by the feature aggregation of the central node according to different nodes and propagate the sentiment information of node to its neighboring nodes.

The set of input and output in the graph attention layer is $h = \{\vec{h}_1, \vec{h}_2, \dots, \vec{h}_N\}$ and $h' = \{\vec{h}'_1, \vec{h}'_2, \dots, \vec{h}'_N\}$, from which the attention coefficient between the central node and neighboring nodes is obtained:

$$e_{ij} = a([Wh_i || Wh'_j]) \quad (8)$$

where a is attention mechanism and W is the weight matrix.

GAT adopts a masked attention mechanism to prevent the dropping of all structural information and changes the previous situation where the self-attention mechanism will allocate attention to all nodes to allocate attention to neighboring nodes. In addition, the attention coefficient is normalized using the *softmax* function, so the attention coefficient after the update is:

$$\alpha_{ij} = \text{softmax}(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k=1}^N \exp(e_{ik})} \quad (9)$$

The multi-head attention mechanism is used to obtain the influence of adjacent nodes on the central node, and the node features extracted by K heads are represented to complete the splicing operation, and finally, the K average is used to replace the connection operation to obtain the final node representation:

$$\vec{h}'_i = \sigma \left(\frac{1}{K} \sum \sum_{j \in N} \alpha_{ij}^k W^k h_j \right) \quad (10)$$

where α_{ij}^k is the normalized attention coefficients and W^k is the linear transformation correlation weight matrix.

By stacking the above update process multiple times, node updates in a multi-layer attention graph can be represented as follows:

$$H_A = \text{MGAT}(A) \quad (11)$$

The syntactic matrix A_{syn} and semantic matrix A_{sem} are fed to the *MGAT*, respectively, to obtain the syntactic feature H_{syn} and semantic feature H_{sem} :

$$H_{syn} = \text{MGAT}(A_{syn}) \quad (12)$$

$$H_{sem} = \text{MGAT}(A_{sem}) \quad (13)$$

3.4. Fusion Layers

Pre-trained language models such as BERT have rich hierarchical information, with phrase-level information at the bottom layer, syntactic feature information in the middle layer, and semantic feature information at the top layer [24]. In addition, according to [29], syntactic and semantic information is not completely isolated, and as the syntactic structure changes, the semantics also have some changes. Interactive learning between syntax and semantics can help us better understand sentences. Therefore, we combine the pre-trained knowledge to fuse and learn the semantic and syntactic information, then feed the output feature H_a into the softmax function for classification, and finally obtain the probability distribution $P(a)$ of the sentiment polarity:

$$H_a = [H_{sem}; H_{syn}; H] \quad (14)$$

$$P(a) = \text{softmax}(W_p H_a + b_p) \quad (15)$$

3.5. Loss Function

We use standard cross-entropy with L_2 as the loss function:

$$L = - \sum_i \sum_{j=1}^C P \log \hat{P} \quad (16)$$

4. Experiment

4.1. Datasets

We evaluate our model on three public datasets: Restaurants and Laptops dataset from Sem-Eval 2014 Task 4 [30] and Twitter dataset provided by Dong et al. [31]. Each sentence in the three datasets is labeled with aspects and opinion words, and sentiment includes three different polarities: positive, neutral, and negative. The statistics from the datasets are in Table 1.

Table 1. Statistics from datasets.

Dataset	Restaurant		Laptop		Twitter	
	Train	Test	Train	Test	Train	Test
Positive	2164	728	994	341	1507	173
Negative	807	196	851	128	1528	169
Neutral	637	196	455	167	3016	336

4.2. Experimental Environment and Parameter Setting

The computing hardware used in the experiment was GeForce GTX 2080Ti, and the deep learning framework was PyTorch. The specific configuration of the experimental environment is shown in Table 2. For model training, we use the bert-base-uncased version of BERT as the sentence encoder and Adam as the optimizer. The detailed parameters are shown in Table 3.

Table 2. Experimental environment.

Projects	Configuration
Operating Platforms	CUDA 11.3
Operating System	Linux
Memory	16 GB
Python Versions	Python 3.8
PyTorch Versions	PyTorch 1.12.0

Table 3. Model parameter settings.

Parameter Name	Parameter Value
batch size	12
learning_rate	0.0001
rnn_hidden	200
bert_dim	768
input_dropout	0.1
atten_head_	2
layer_dropout	0.2
num_epoch	20
attn_head	2

4.3. Evaluation Index

Following the previous work, we used Accuracy and Macro-F1 values as evaluation indexes of aspect-level sentiment analysis tasks.

4.4. Baseline Methods

We selected some mainstream baseline and latest models to compare with the proposed models.

- (1) IAN [8]: The aspect words and contextual representations generated by LSTM are used to learn interactively through attention.
- (2) AOA [10]: The aspect words and context representations generated by LSTM are modeled by attention-over-attention neural networks to capture the interaction between aspect and context.
- (3) RAM [9]: This proposes a recurrent attention network on memory to capture sentiment features between long distances.
- (4) MGAN [11]: The alignment matrix is used to complete the coarse-grained interaction between the aspect word and the context, and the aspect alignment loss function is designed to complete the fine-grained interaction at the word level.
- (5) TNet [32]: Use CNN to extract significant features from the transformed word representations from the bidirectional RNN layer.
- (6) ASGCN [12]: The dependency tree is used to extract syntactic information and perform graph convolution operations on the dependency tree to learn the representation of nodes.
- (7) CDT [13]: The feature representation of a sentence is learned by using bidirectional LSTM, and the embedded representation is enhanced by graph convolutional networks.
- (8) BiGCN [14]: The hierarchical syntactic graph and lexical graph are constructed by convolution on GNN embedding and BiLSTM embedding, respectively, and a bi-level interactive network is designed to learn information interaction.
- (9) kumaGCN [15]: It combines information from the latent graph and the dependency graph through a gated attention mechanism.
- (10) R-GAT [16]: The dependency tree is rooted to the target aspect by reconstructing, and pruning is performed to preserve the edges that are directly dependent on the aspect term.
- (11) DGEDT [15]: Considering the dependency tree as a special form of transformer, representations from the dependency tree and transformer are learned in an iterative interaction manner.
- (12) DualGCN [26]: Syntactic graph and semantic graph are constructed at the same time, and a double affine mechanism is used to complete the information exchange between syntactic and semantic, and finally, all the information is fused for classification.
- (13) SSEGCN [2]: The attention matrix constructed by the attention mechanism and syntactic mask matrix are combined to accomplish the interaction of syntactic structure and semantic information.
- (14) RAG-TCGCN [33]: Multiple attention is used to combine syntactic and semantic features and their related features with word-level features parsed using residual attention gates.
- (15) BERT [21]: MLM is used for pre-training bidirectional transformers to generate deep language representation, and good results can be achieved only with fine-tuning in downstream tasks.
- (16) BERT-PT [22]: The pre-training language model is trained through a large number of general domain data and a small amount of downstream data. It provides a solution for small sample data research.
- (17) AEN-BERT [23]: This uses an attention encoding network to interact aspect words with context and designs a processing form based on BERT word embedding.
- (18) BERT4GCN [25]: Based on BERT's rich hierarchical structure information, the feature information in the middle layer is fused with the knowledge of the dependency tree, the enhanced dependency graph is constructed, and the convolution operation is performed in it.

4.5. Experimental Results and Analysis

Our proposed model is compared with three types of baseline model: the attention-based method, the syntax-based model, and the pre-training-based model. The attention-based model includes IAN, AOA, RAM, MGAN, and TNet. The syntax-based model includes ASGCN, CDT, BiGCN, kumaGCN, R-GAT, DGEDT, DualGCN, SSEGCN, and RAG-TCGCN. The pre-training-based model includes BERT, BERT-PT, AEN-BERT, and BERT4GCN. The main experimental results are reported in Table 4.

Table 4. Sentiment classification results. We directly introduce the result data from the original author’s paper as the data for comparison, where “-” means that this part of the work is not revealed, and the best experimental results are shown in bold.

Models	Restaurant		Laptop		Twitter	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
IAN	78.60	-	72.10	-	-	-
AOA	80.53	69.84	72.88	67.48	72.25	69.96
RAM	80.23	70.80	74.49	71.35	69.36	67.30
MGAN	81.25	71.94	75.39	72.47	72.54	70.81
TNet	80.69	71.27	76.54	71.75	74.90	73.60
ASGCN-DG	80.77	72.02	75.55	71.05	72.15	70.40
ASGCN-DT	80.86	72.19	74.14	69.24	71.53	69.68
CDT	82.30	74.02	77.19	72.99	74.66	73.66
BiGCN	81.97	73.48	74.59	71.84	74.16	73.35
kumaGCN	81.43	73.64	76.12	72.42	72.45	70.77
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
SSEGCN	84.72	77.51	79.43	76.49	76.51	75.32
RAG-TCGCN	84.09	77.02	78.80	75.04	76.66	75.41
BERT-PT	84.95	76.96	78.07	75.08	-	-
BERT-SPC	84.46	76.98	78.99	75.03	73.55	72.14
AEN-BERT	83.12	73.76	79.93	76.31	74.71	73.13
BERT4GCN	84.75	77.11	77.49	73.01	74.73	73.76
Ours	86.42	79.70	80.06	76.78	76.81	76.10

Based on the experimental results in Table 4, we offer the following analysis:

- (1) Our proposed model achieves better results compared with other last and baseline models. We believe that the primary reason is that the designed SSEMGAT model captures syntactic and semantic feature information more efficiently than other models, which also proves the effectiveness of our work.
- (2) The model that considers syntactic structure and semantic information at the same time is better than the model that considers only semantic information or syntactic structure, which shows that syntax and semantics do not exist in isolation, and learning the interaction information between them is also very necessary.
- (3) Compared with attention-based models, our proposed model has obvious advantages. From the analysis of this phenomenon, we believe that the attention mechanism is easily affected by the noise factor in the sentence when facing complex sentences and obscure structures and cannot accurately align the contextual and sentiment information. This reduces the performance of the model.
- (4) Compared with syntax-based models, our model also has good results. This may be because we made up for the inherent defects in dependency trees in sentence parsing, thus enhancing their ability to capture aspect words and their corresponding opinion words and improving the model’s ability to resist interference to noise elements introduced in the dependency tree.
- (5) Compared with the model based on pre-training, our model also has better performance. BERT has strong representational learning ability and a rich hierarchical structure, while the dependency tree also has an obvious hierarchical structure, which

may be related in some way. When we use the enhanced feature extractor for extraction, we can better capture the correlation between syntax and semantics.

4.6. Ablation Study

We further conducted an ablation study to verify the validity of each module in our model. The result is in Table 5. In the ablation experiment, we removed the dependency tree (dep), constituent tree (con), aspect-aware attention (aaa), and multi-head attention (mha) for comparison and verification.

Table 5. The results of the ablation study.

	Restaurant		Laptop		Twitter	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
SSEMGAT	86.42	79.70	80.06	76.78	76.81	76.10
w/o dep	85.69	78.69	79.15	75.97	74.26	73.56
w/o con	85.52	78.16	78.80	74.67	75.62	74.43
w/o aaa	86.05	79.66	79.75	76.00	76.22	75.70
w/o mha	85.25	78.02	78.31	75.73	75.31	75.22

First, removal of the dependency tree (dep) leads to a drop in accuracy of 0.73%, 0.91%, and 2.65% on the Restaurant, Laptop, and Twitter dataset, respectively, which demonstrates that the dependency tree is important for extracting syntactic information. Then, with the removal of the constituent tree (w/o con), the model performance decreases by 0.9%, 1.26%, and 1.19%, respectively. It is shown that the constituent tree can effectively supplement the syntactic information extracted from the dependency tree. After, the removal of aspect-aware attention (w/o aaa) causes a decay in the accuracy of 0.37%, 0.31%, and 0.59%. As for ‘w/o mha’, the accuracy decreases by 1.17%, 1.75%, and 1.5% on the Restaurants, Laptop, and Twitter datasets, respectively. As a result, the ablation experimental outcomes confirm the contribution of both components.

4.7. Case Study

To better understand the work of the SSEMGAT model, we selected two samples to review for visual case studies. In Table 6, we visualize the attention weights, predicted labels, aspect terms, and corresponding true labels for sentences.

Table 6. Visual analysis of attention in review sample.

Model	Aspect	Attention Visualization	Prediction	Label
AOA	environment	Its environment is elegant but price is expensive	Negative	Positive
	price	Its environment is elegant but price is expensive	Positive	Negative
	room	The look of the room is novel	Positive	Positive
ASGCN	environment	Its environment is elegant but price is expensive	Negative	Positive
	price	Its environment is elegant but price is expensive	Negative	Negative
	room	The look of the room is novel	Positive	Positive
BERT-BASE	environment	Its environment is elegant but price is expensive	Positive	Positive
	price	Its environment is elegant but price is expensive	Positive	Negative
	room	The look of the room is novel	Positive	Positive
SSEMGAT	environment	Its environment is elegant but price is expensive	Positive	Positive
	price	Its environment is elegant but price is expensive	Negative	Negative
	room	The look of the room is novel	Positive	Positive

The first sample contains two aspect terms where the corresponding sentiment polarity is opposite, and the second sample contains only one aspect term.

In the first example, the AOA model focuses on “elegant” and “but” at the same time, misjudges “environment” as negative sentiment polarity, while “price” focuses on “elegant” and “expensive” and allocates positive sentiment polarity. This shows that there is interference between different aspect terms. In the second example, with only one aspect term, the correct sentiment polarity was identified. The ASGCN model may misjudge sentiment by taking the relationship between “but” and “environment” as clues. The BERT model does not correctly align the sentiment information corresponding to “price”. We speculate that the possible reason is that the corresponding sentiment words are randomly replaced with other irrelevant information when masking. The SSEMGAT model effectively combined syntactic structure and semantic correlation of the feature information and correctly predicted all aspects of terms related to sentiment tendency.

5. Conclusions and Future Work

In this paper, we proposed a syntactic and semantic enhanced multi-layer graph attention neural network (SSEMGAT) to solve the problem of introducing noise in dependent trees in sentences without obvious syntactic structure. Given the inherent defects in dependent trees, we introduced the composition tree structure, which can obtain more field-of-view information at the causal level, and we enhanced the syntactic features by merging syntactic information at different levels. The multi-head attention mechanism may misjudge the sentiment polarity due to the noise introduced by the interference of irrelevant words, so we construct local attention and global attention of specific aspects based on the attention mechanism to assign the attention weight between aspect and context. Facing feature information with a rich hierarchy, we used the multi-layer stacked graph attention module to aggregate different hierarchical information separately and used attention to give higher weight to the information most relevant to the feature. Finally, the extracted syntactic and semantic features are fused with the pre-training knowledge to obtain the most specific aspect of rich hierarchical feature information to achieve aspect sentiment classification.

In future research, we will continue to apply the model to different domains to verify the generalization performance and observe the model’s performance in multilingual datasets. Current research still has challenges in mining deeper correlation information between syntax and semantics, and we will further develop methods that can dig deeper into the correlation between them.

Author Contributions: Conceptualization, X.X. and A.W.; methodology, X.X.; software, J.H.; validation, X.X.; formal analysis, X.X., A.W. and Z.K.; investigation, J.H.; data curation, A.W.; writing—original draft preparation, X.X. and A.W.; writing—review and editing, X.X.; supervision, J.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Autonomous Region Natural Science Foundation Joint Fund Project, Research on Xinjiang Tourism Sentiment Analysis Technology Based on Deep Learning, under Grant 2021D01C081.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. 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 of the Association for Computational Linguistics: ACL 2022*; Association for Computational Linguistics: Dublin, Ireland, 2022; pp. 1835–1848.
2. Zhang, Z.; Zhou, Z.; Wang, Y. SSEGCN: Syntactic and Semantic Enhanced Graph Convolutional Network for Aspect-Based Sentiment Analysis. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Seattle, WA, USA, 10–15 July 2022; Association for Computational Linguistics: Dublin, Ireland, 2022; pp. 4916–4925.
3. Yang, B.; Cardie, C. Context-Aware Learning for Sentence-Level Sentiment Analysis with Posterior Regularization. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Baltimore, MD, USA, 23–24 June 2014; Association for Computational Linguistics: Dublin, Ireland, 2014; pp. 325–335.
4. Severyn, A.; Moschitti, A. Twitter Sentiment Analysis with Deep Convolutional Neural Networks. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, 9–13 August 2015; Association for Computing Machinery: New York, NY, USA, 2015; pp. 959–962.
5. Dou, Z.-Y. Capturing User and Product Information for Document Level Sentiment Analysis with Deep Memory Network. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017; Association for Computational Linguistics: Dublin, Ireland, 2017; pp. 521–526.
6. Lyu, C.; Foster, J.; Graham, Y. Improving Document-Level Sentiment Analysis with User and Product Context. In Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain, 8–13 December 2020; pp. 6724–6729.
7. Wang, Y.; Huang, M.; Zhu, X.; Zhao, L. Attention-Based LSTM for Aspect-Level Sentiment Classification. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; Association for Computational Linguistics: Dublin, Ireland, 2016; pp. 606–615.
8. Ma, D.; Li, S.; Zhang, X.; Wang, H. Interactive Attention Networks for Aspect-Level Sentiment Classification. *arXiv* **2017**, arXiv:1709.00893.
9. Chen, P.; Sun, Z.; Bing, L.; Yang, W. Recurrent Attention Network on Memory for Aspect Sentiment Analysis. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 7–11 September 2017; Association for Computational Linguistics: Dublin, Ireland, 2017; pp. 452–461.
10. Huang, B.; Ou, Y.; Carley, K.M. Aspect Level Sentiment Classification with Attention-over-Attention Neural Networks. In Proceedings of the Social, Cultural, and Behavioral Modeling: 11th International Conference, SBP-BRiMS 2018, Washington, DC, USA, 10–13 July 2018; Thomson, R., Dancy, C., Hyder, A., Bisgin, H., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 197–206.
11. Fan, F.; Feng, Y.; Zhao, D. Multi-Grained Attention Network for Aspect-Level Sentiment Classification. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; Association for Computational Linguistics: Dublin, Ireland, 2018; pp. 3433–3442.
12. Zhang, C.; Li, Q.; Song, D. Aspect-Based Sentiment Classification with Aspect-Specific Graph Convolutional Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics: Dublin, Ireland, 2019; pp. 4568–4578.
13. Sun, K.; Zhang, R.; Mensah, S.; Mao, Y.; Liu, X. Aspect-Level Sentiment Analysis via Convolution over Dependency Tree. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics: Dublin, Ireland, 2019; pp. 5679–5688.
14. Zhang, M.; Qian, T. Convolution over Hierarchical Syntactic and Lexical Graphs for Aspect Level Sentiment Analysis. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online. 16–20 November 2020; Association for Computational Linguistics: Dublin, Ireland, 2020; pp. 3540–3549.
15. Chen, C.; Teng, Z.; Zhang, Y. Inducing Target-Specific Latent Structures for Aspect Sentiment Classification. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online. 19–20 November 2020; Association for Computational Linguistics: Dublin, Ireland, 2020; pp. 5596–5607.
16. Wang, K.; Shen, W.; Yang, Y.; Quan, X.; Wang, R. Relational Graph Attention Network for Aspect-Based Sentiment Analysis. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online. 6–8 July 2020; Association for Computational Linguistics: Dublin, Ireland, 2020; pp. 3229–3238.
17. Tian, Y.; Chen, G.; Song, Y. Aspect-Based Sentiment Analysis with Type-Aware Graph Convolutional Networks and Layer Ensemble. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online. 6–11 June 2021; Association for Computational Linguistics: Dublin, Ireland, 2021; pp. 2910–2922.

18. Li, R.; Chen, H.; Feng, F.; Ma, Z.; Wang, X.; Hovy, E. Dual Graph Convolutional Networks for Aspect-Based Sentiment Analysis. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online. 1–6 August 2021; Association for Computational Linguistics: Dublin, Ireland, 2021; pp. 6319–6329.
19. Wu, H.; Zhang, Z.; Shi, S.; Wu, Q.; Song, H. Phrase Dependency Relational Graph Attention Network for Aspect-Based Sentiment Analysis. *Knowl.-Based Syst.* **2022**, *236*, 107736. [[CrossRef](#)]
20. Wang, X.; Zhu, M.; Bo, D.; Cui, P.; Shi, C.; Pei, J. AM-GCN: Adaptive Multi-Channel Graph Convolutional Networks. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Long Beach, CA USA, 6–10 July 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 1243–1253.
21. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, MN, USA, 2–7 June 2019; Association for Computational Linguistics: Dublin, Ireland, 2019; pp. 4171–4186.
22. Xu, H.; Liu, B.; Shu, L.; Yu, P. BERT Post-Training for Review Reading Comprehension and Aspect-Based Sentiment Analysis. In 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), Minneapolis, MN, USA, 2–7 June 2019; Association for Computational Linguistics: Dublin, Ireland, 2019; pp. 2324–2335.
23. Song, Y.; Wang, J.; Jiang, T.; Liu, Z.; Rao, Y. Attentional Encoder Network for Targeted Sentiment Classification. *arXiv* **2019**, arXiv:1902.09314.
24. Jawahar, G.; Sagot, B.; Seddah, D. What Does BERT Learn about the Structure of Language? In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; Association for Computational Linguistics: Dublin, Ireland, 2019; pp. 3651–3657.
25. Xiao, Z.; Wu, J.; Chen, Q.; Deng, C. BERT4GCN: Using BERT Intermediate Layers to Augment GCN for Aspect-Based Sentiment Classification. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Punta Cana, Dominican Republic, 7–11 November 2021; Association for Computational Linguistics: Dublin, Ireland, 2021; pp. 9193–9200.
26. Tang, H.; Ji, D.; Li, C.; Zhou, Q. Dependency Graph Enhanced Dual-Transformer Structure for Aspect-Based Sentiment Classification. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online. 6–8 July 2020; Association for Computational Linguistics: Dublin, Ireland, 2020; pp. 6578–6588.
27. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2017**, arXiv:1706.03762.
28. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks. *arXiv* **2018**, arXiv:1710.10903.
29. Pyllkkänen, L. The Neural Basis of Combinatory Syntax and Semantics. *Science* **2019**, *366*, 62–66. [[CrossRef](#)] [[PubMed](#)]
30. Pontiki, M.; Galanis, D.; Pavlopoulos, J.; Papageorgiou, H.; Androutsopoulos, I.; Manandhar, S. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, 23–24 August 2014; Association for Computational Linguistics: Dublin, Ireland, 2014; pp. 27–35.
31. Dong, L.; Wei, F.; Tan, C.; Tang, D.; Zhou, M.; Xu, K. Adaptive Recursive Neural Network for Target-Dependent Twitter Sentiment Classification. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Baltimore, MD, USA, 22–27 June 2014; Association for Computational Linguistics: Dublin, Ireland, 2014; pp. 49–54.
32. Li, X.; Bing, L.; Lam, W.; Shi, B. Transformation Networks for Target-Oriented Sentiment Classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 15–20 July 2018; Association for Computational Linguistics: Dublin, Ireland, 2018; pp. 946–956.
33. Xu, H.; Liu, S.; Wang, W.; Deng, L. RAG-TCGCN: Aspect Sentiment Analysis Based on Residual Attention Gating and Three-Channel Graph Convolutional Networks. *Appl. Sci.* **2022**, *12*, 12108. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.