DAGCN: Distance-based and Aspect-oriented Graph Convolutional Network for Aspect-based Sentiment Analysis

Zhihao Wang[1], **Bo Zhang**[1,2]*, **Ru Yang**[1]†, **Chang Guo**[1], **Mao-Zhen Li**[1,3]
[1]College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, China
[2]Shanghai Engineering Research Center of Intelligent Education and Bigdata, Shanghai Normal University,China
[3]Department of Electronic and Electrical Engineering, Brunel University London, Uxbridge, UK
zhihaolancorrect@gmail.com
{zhangbo, yangru, guochang}@shnu.edu.cn
maozhen.li@brunel.ac.uk

Abstract

Aspect-based sentiment analysis (ABSA) is a task that aims to determine the sentiment polarity of aspects by identifying opinion words. Recent advancements have predominantly been rooted either in semantic or syntactic methods. However, both of them tend to interference from local factors such as irrelevant words and edges, hindering the precise identification of opinion words. In this paper, we present Distance-based and Aspect-oriented Graph Convolutional Network (DAGCN) to address the aforementioned issue. Firstly, we introduce the Distance-based Syntactic Weight (DSW). It focuses on the local scope of aspects in the pruned dependency trees, thereby reducing the candidate pool of opinion words. Additionally, we propose Aspect-Fusion Attention (AF) to further filter opinion words within the local context and consider cases where opinion words are distant from the aspect. With the combination of DSW and AF, we achieve precise identification of corresponding opinion words. Extensive experiments on three public datasets demonstrate that the proposed model outperforms state-of-the-art models and verify the effectiveness of the proposed architecture.

1 Introduction

Aspect-based sentiment analysis (ABSA) is a fine-grained sentiment analysis task that aims to determine the sentiment polarity of a given aspect within a sentence. The sentiment polarity can be classified into three categories: positive, neutral, and negative. For instance, in Figure 1, the aspect *"skype"* can be determined to have positive sentiment polarity based on opinion word *"cool"*. In fact, opinion words carry certain sentiment information and ABSA primarily focuses on identifying opinion words that are relevant to the aspect.

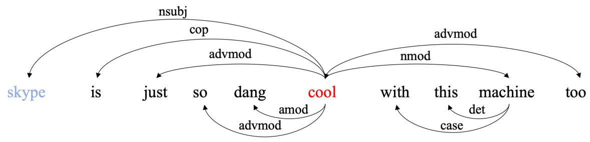


Figure 1: An example sentence with its dependency tree. There are two aspects (bolded in black) in this sentence but these aspects contain opposite sentiment polarities.

Previous studies have explored heavily on attention mechanism methods and achieved promising results (Chen et al., 2017; Ma et al., 2017a; Nguyen and Le Nguyen, 2018; Liu et al., 2018; Ma et al., 2018; Mokhosi et al., 2019). In these works, Attention mechanism is utilized to model the correlation between aspects and context words. However, they always suffer noise that high weights might be given wrongly to words that are irrelevant to the aspect.

For the purpose of filtering out the noise brought by the attention mechanism, Semantic-Relative Distance (SRD) is proposed to measure semantic correlation degree (Zeng et al., 2019). It could help attention mechanism identify opinion words more accurately in local range. Therefore, a plenty of studies (Liu et al., 2022; Yu and Zhang, 2023) utilized SRD or its variants to improve ABSA task. However, this local scope is still large, and it also tends to recognize other words as opinion words mistakenly. To prove it, we conduct a statistical analysis of SRD with the datasets, Lap14 and Res14, provided by (Fan et al., 2019). Table 1 shows that the data with SRD <= 2 accounts for approximately 51% of the total dataset. While the data with SRD <= 6 essentially constitutes the majority of the dataset.

On the other hand, more significant efforts (Tang et al., 2020; Chen et al., 2021; Yan et al., 2021; Li et al., 2021b; Tang et al., 2022; Zhang et al., 2022; Zhong et al., 2023) in ABSA have focused on dependency tree, which has ability to analyze syntactic structure from the grammatical perspec-

* Corresponding author
† Corresponding author

Findings of the Association for Computational Linguistics: NAACL 2024, pages 1863–1876

Table 1: Statistical ratio of syntactic distance and semantic-related distance.

Dataset	Category	<=2	<=4	<=6
Lap14	SD	83.33%	97.27%	99.25%
	SRD	51.41%	75.05%	84.84%
Res14	SD	85.86%	98.13%	99.60%
	SRD	51.30%	75.85%	87.93%

tive. Subsequently, GCNs and GATs aggregate node features over the adjacency matrix derived from the dependency tree to determine the sentiment polarity of the aspect. However, dependency tree just reveals whether syntactic connection exists between context words and aspects, and it's hard to distinguish which words are valuable.

To tackle the issues above, we would like to reduce noise in dependency tree just like SRD and focus on opinion words precisely. Some studies (He et al., 2018; Zhou et al., 2021; Chen et al., 2022) have shown that in most cases opinion words are close to the aspects in a dependency tree, which means that we can only consider context words surrounding aspects syntactically. Figure 2 demonstrates that syntactic distance (SD) between opinion words and aspects in most cases is shorter than 2 and the data with SD larger than 6 comprises an extremely small portion. Besides, Table 1 shows that the amount of data for SD is obviously larger than the amount of data for SRD when their values are equal. This statistical result indicates that SD could better reflect that context words surrounding aspects are more likely to be opinion words.

Therefore, we propose Distance-based Syntactic Weight (DSW) computed through Aspect-Oriented Dependency Tree (Wang et al., 2020). Note that AODT is utilized from the perspective of SD in our paper, rather than dependency relationships. DSW characterizes the syntactic correlation strength between context words and aspects, and enhances the precision of identifying opinion words. Then, we define Distance-based Weighted Matrix (DWM) to store DSW. Considering that opinion words are far away from aspects, we introduce Aspect-Fusion attention (AF) to further discern candidate opinion words within both local and long range scopes. Finally, we combine DWM and AF to build an adjacency matrix and a Graph Convolutional Network (AoGCN) is constructed over it.

Due to the fact that GCNs over dependency tree perform poorly on the reviews with informal ex-

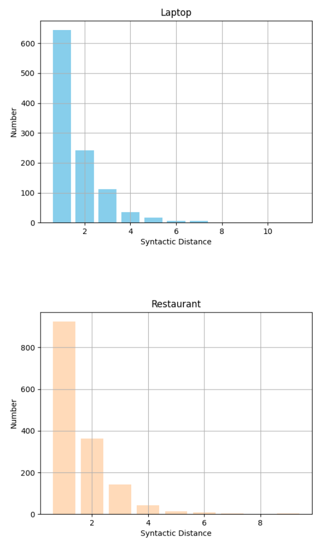


Figure 2: Syntactic distance statistics in Lap14 and Res14.

pression, similar to DualGCN (Li et al., 2021a), we build another GCN (SaGCN) by employing self-attention mechanism. Specifically, we incorporate a Kullback-Leibler (KL) divergence loss to ensure that the two GCNs learn distinct features, with AoGCN focusing on syntactic information and SaGCN emphasizing semantic information. Main contributions are summarized as follows:

- We propose DSW to augment the precision in discerning opinion words within a local scope and more precisely elevate the contribution of opinion words to ABSA task.
- We present AF to account for situations where opinion words are distant from aspects. It remedies the local-centric focus and overlook of the global context.
- We conduct experiments on the SemEval 2014 and Twitter datasets, and achieved state-of-the-art results, validating the effectiveness of the DAGCN architecture. To facilitate the reproducibility of our work, datasets and the source code are provided on GitHub[1].

2 Related Work

Aspect-based sentiment analysis primarily focuses on utilizing opinion words to determine the sentiment polarity of aspects. Early works (Thelwall and Buckley, 2013; Kim et al., 2013) often

[1] https://github.com/lancorrect/DAGCN.git

relied on constructing aspect-specific sentiment lexicons or manually specified features, without incorporating syntactic features.

Recently, lots of works have focused extensively on attention mechanism to determine the semantic correlation between context words and aspects (Wang et al., 2016b; Chen et al., 2017; Ma et al., 2017a; Nguyen and Le Nguyen, 2018; Liu et al., 2018; Ma et al., 2018; Mokhosi et al., 2019; Deng et al., 2019). Ma et al. (2018) designed stacked attention mechanisms to capture both local and global features, enhancing the performance of LSTM. Deng et al. (2019) proposed a novel sparse self-attention mechanism to differentiate the importance of different words for sentiment polarity.

There have been several studies focusing on the distance between aspects and opinion words, as it is believed to contain rich semantic knowledge (Zeng et al., 2019). In addition to utilizing SRD to extract semantic information, Liu et al. (2022) also used the absolute distance between aspects and context words to differentiate their importance. Yu and Zhang (2023) created a local context weighted adjacency graph with SRD in order to emphasize significance of local context and avoid long range influence. However, these methods were hard to precisely identify opinion words within relatively large local scopes.

In addition, the dependency tree has been widely used before. Nguyen and Shirai (2015) integrated syntactic information by combining dependency relation and phrases. Wang et al. (2016a) utilized underlying syntactic information to learn a high-level feature representation. With the emergence of Graph Convolutional Networks (GCNs) and Graph Attention Networks (GATs), GCN-based and GAT-based methods have been employed to learn syntactic information from the dependency tree (Zhang et al., 2019; Sun et al., 2019; Wang et al., 2020; Tang et al., 2020; Zhang et al., 2022; Chen et al., 2021; Yan et al., 2021; Li et al., 2021b; Tang et al., 2022; Zhong et al., 2023; Jiang et al., 2023). Li et al. (2021b) selected relevant knowledge from a knowledge graph and incorporated it into the dependency tree to improve its expressive power. Tang et al. (2022) considered the relationship labels of the dependency tree and proposed an adaptive fusion module for semantic information. Unfortunately, the dependency tree contained considerable noise, with non-opinion words interfering with the model's judgment. Additionally, these methods failed to recognize the contribution of opinion words to the ABSA task.

3 The Proposed model

Figure 3 illustrates the overview of DAGCN. Given a pair of sentence-aspect (s, a), where $s = \{w_1, w_2, ..., w_n\}$ and $a = \{a_1, a_2, ..., a_m\}$, an aspect is a part of the sentence. m is the end position of aspect in s. Before feeding the input into the model, we first map each word to its embedding with the embedding table $E \in \mathbb{R}^{|V| \times d_e}$, where $|V|$ represents the size of the embedding table and d_e denotes the dimension of the word embeddings. Then, an encoder such as BiLSTM or BERT is utilized to learn contextual information from the sentence. The input x is fed into the encoder, resulting in hidden state vectors $H = \{h_1, h_2, ..., h_n\}$, where $h_i \in \mathbb{R}^{2d_h}$ and $2d_h$ represents the dimension of the hidden state vectors obtained from the encoder. We use H as the initial node representation and input it into AoGCN and SaGCN for aggregation operations. For the BERT encoder, we construct inputs in the format required by BERT, which is "[CLS] sentence [SEP] aspect [SEP]". [CLS] and [SEP] are special tokens in BERT used for classification and sentence separation, respectively. Subsequently, we elaborate on the details of DAGCN.

3.1 Distance-based Weighted Matrix (DWM)

Algorithm 1 Distance-based Weighted Matrix

Require: aspect $a = \{a_1, a_2, ..., a_m\}$, sentence $s = \{w_1, w_2, ..., w_n\}$, positions $p = \{k, k+1, ..., k_{m-1}\}$, dependency Tree T
Ensure: Distance-based Weighted Matrix M

```
Initialize a zero initialization matrix M
Convert T into AODT
for i = 1 to m do
    for j = 1 to n do
        if a_i and w_j are directly connected then
            SD = 1
        else
            SD = DFS(a_i, w_j)
        end if
        DSW = exp(α · SD)
        M_{p[i-1]j} = DSW, M_{jp[i-1]} = DSW
    end for
end for
return M
```

Given that the primary focus of ABSA is on aspects, edges in the dependency tree not directly

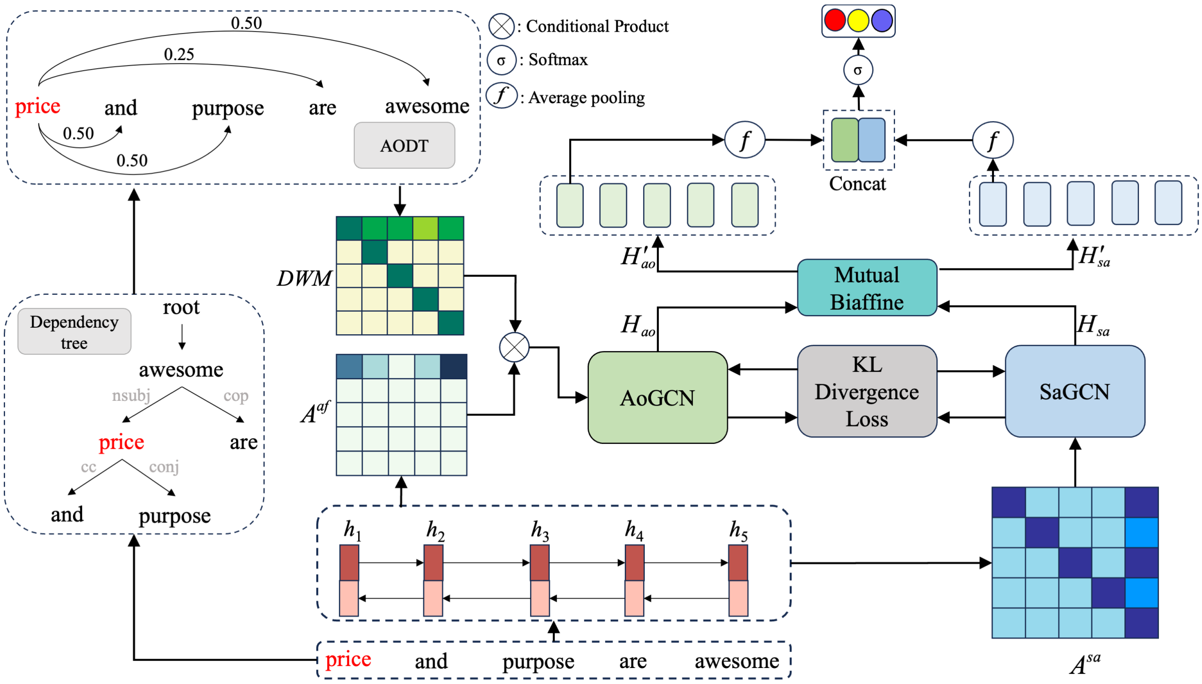


Figure 3: The overall architecture of DAGCN. Values over the edges in AODT represent DSW.

linked to the aspect are perceived to offer limited assistance in prediction. Therefore, it's necessary to prune the dependency tree and reduce noise. Based on this issue, we propose Distance-based Syntactic Weight and construct DWM.

Algorithm 1 describes the construction process of DWM. For an input sentence, we use a dependency parser to perform syntactic analysis and generate a dependency tree. Followed by R-GAT(Wang et al., 2020), Aspect-Oriented Dependency Tree (AODT) eliminates all edges not directly linked to aspects and prioritizes GCN's focus on the aspect's local context in the dependency tree. However, in contrast to AODT, when context words are not directly linked to aspects, we use the result of depth-first search (DFS) to represent the distance between them, rather than using virtual relation to depict their connection. Note that maybe the context words are not linked to aspects at all. In such cases, we set SD between words to infinity. Additionally, if an aspect consists of multiple words, we need to calculate SD between each word in an aspect and the context words separately.

Then, we multiply SD by a scalar α ($\alpha < 0$) and apply the exponential function $exp()$ to obtain DSW. If SD between the context words and aspects is large, DSW is close to 0, indicating that the context words are negligible for the aspect. If SD between context words and aspects is smaller, DSW becomes larger, indicating that context words within the local scope of the aspect contribute significantly to determining the sentiment polarity. At this stage, each DSW ranges from 0 to 1. A zero initialization matrix $M \in \mathbb{R}^{n\times n}$ is built as DWM to store DSWs and $p = \{k, k+1, ..., k_{m-1}\}$ represents the aspect positions in s. With DSW, opinion words can be more efficiently and accurately identified, stand out from other unrelated context words.

3.2 Aspect-Fusion Attention (AF)

While we have narrowed down the candidate range for opinion words, non-opinion words still exist within this scope, and there are few instances where opinion words have a substantial syntactic distance from aspects. Therefore, we introduce AF and the computation process is described as follows:

$$A^{af} = avg(tanh(H_a W^a_{af} \times (K_{af} W^K_{af})^T + b)) \quad (1)$$

Where K_{af} is the output H of the encoder. $W^a_{af} \in \mathbb{R}^{2d_h\times 2d_h}$ and $W^K_{af} \in \mathbb{R}^{2d_h\times 2d_h}$ are learnable weights. Note that H_a is obtained from H by keeping only the word embeddings at the aspect positions, i.e., $H_a = \{0, 0, .., h_{a_1}, h_{a_2}, ..., h_{a_m}, ..., 0\}$, $H_a \in \mathbb{R}^{n\times 2d_h}$. In fact, as an aspect may comprise multiple words, AF considers each word in the aspect as a query to compute attention scores with context words. With AF, the aspect could distinguish important words for itself in short or long distance. Average pooling, denoted as $avg()$, is then utilized to average attention scores for regrading them as a whole. Finally, a zero-initialization matrix $A^{af} \in \mathbb{R}^{n \times n}$ is constructed and the output vector is copied to the aspect positions in A^{af}.

3.3 Aspect-oriented GCN (AoGCN)

In order to incorporate local and global syntactic information, we combine DWM with AF. The process of fusion is defined as follows:

$$M_{ij} = \begin{cases} 1, & AF_{ij} > \beta \\ M_{ij}, & otherwise \end{cases} \quad (2)$$

Where M_{ij} indicates the corresponding value of DWM between w_i and w_j. AF_{ij} represents the attention weight between w_i and w_j in A^{af}. β is a hyperparameter ($\beta > 0$).

When AF_{ij} is higher than β, it indicates that the corresponding context word is highly important for the aspect, and its distance weight in DWM needs to be increased to the maximum value. Conversely, when AF_{ij} is smaller than or equal to β, it suggests that the context word contributes less to the prediction, and its distance weight remains unchanged. Then, we multiply AF by DWM to obtain the adjacency matrix for AoGCN:

$$A^{ao} = A^{af} \times M \quad (3)$$

Where $A^{ao} \in \mathbb{R}^{n \times n}$. A^{ao} ensures the identification of the most probable opinion words, whether in local or global contexts. Specially, we treat the element of A^{ao} as Comprehensive Syntactic Value (CSV), which explain the significance of context words from the perspective of overall syntax.

Based on A^{ao}, we can build AoGCN. Assume that the input to the l-th layer is h^{l-1} and the output is h^l. The initial input is h^0. In the l-th layer, the hidden state h_i^l of the i-th node can be updated by aggregating the hidden states of its neighboring nodes through the following operation:

$$h_i^l = \sigma(\sum_{j=1}^{n} A^{ao} W^l h_j^{l-1} + b^l) \quad (4)$$

where W^l and b^l are learnable weight matrix and bias, respectively. σ is a non-linear activation function. The output of AoGCN in the last layer is denoted as $H_{ao} = \{h_1^{ao}, h_2^{ao}, ..., h_n^{ao}\}$, where h_i^{ao} represents the hidden state of word w_i in the last layer of AoGCN.

3.4 Self-attention GCN (SaGCN)

Similar with DualGCN (Li et al., 2021a), another GCN (SaGCN) is built with self-attention mechanism. It prioritizes semantic features and greatly assists in sentences with unclear syntactic structures. The attention scores between every pair of words indicates the level of semantic correlation. The calculation is shown as followed:

$$A^{sa} = \frac{QW_{sa}^{Q} \times (KW_{sa}^{K})^T}{\sqrt{d_h}} \quad (5)$$

Where Q and K are the same as the input of the l-th layer, which is h^{l-1}. $W_{sa}^{Q} \in \mathbb{R}^{2d_h \times 2d_h}$ and $W_{sa}^{K} \in \mathbb{R}^{2d_h \times 2d_h}$ are learnable weight matrices. Similar to AoGCN, SaGCN ultimately obtains the graph representation H_{sa}.

3.5 BiAffine Module

To effectively interact the features learned by AoGCN and SaGCN, we employ a mutual BiAffine transformation (Tang et al., 2020) as an intermediate exchange:

$$\begin{aligned} H'_{ao} &= softmax(H_{ao} W_1 (H_{sa})^T) H_{sa} \\ H'_{sa} &= softmax(H_{sa} W_2 (H_{ao})^T) H_{ao} \end{aligned} \quad (6)$$

where W_1 and W_2 are learnable parameters. H'_{ao} and H'_{sa} represent the output results of the BiAffine Module, respectively. Suppose that the aspect nodes in H'_{ao} are represented by $\{h_{a_1}^{ao}, h_{a_2}^{ao}, ..., h_{a_m}^{ao}\}$ and in H'_{sa} by $\{h_{a_1}^{sa}, h_{a_2}^{sa}, ..., h_{a_m}^{sa}\}$. Then, we can obtain the final representation of aspects through the following calculation:

$$h_a^{ao} = f(h_{a_1}^{ao}, h_{a_2}^{ao}, ..., h_{a_m}^{ao}) \quad (7)$$

$$h_a^{sa} = f(h_{a_1}^{sa}, h_{a_2}^{sa}, ..., h_{a_m}^{sa}) \quad (8)$$

$$h_f = [h_a^{ao}, h_a^{sa}] \quad (9)$$

Where $f(\cdot)$ represents average pooling and $[\cdot]$ denotes concatenation operation. Next, we input the final representation of aspects into a linear layer, and then the output passes through a $softmax()$ function to obtain a probability distribution vector for sentiment polarity:

$$p(a) = softmax(W_f h_f + b_f) \quad (10)$$

Where W_f and b_f are learnable weight matrix and bias.

Table 2: Experimental results comparison on three publicly benchmark datasets.

Models	Restaurant		Laptop		Twitter	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
IAN(Ma et al., 2017b)	78.6	-	72.10	-	-	-
RAM(Chen et al., 2017)	80.23	70.80	74.49	71.35	69.36	67.30
TNet(Li et al., 2018)	80.69	71.27	76.54	71.75	74.90	73.60
LCF(Zeng et al., 2019)	82.50	73.92	76.02	70.58	72.25	70.92
ASGCN(Zhang et al., 2019)	80.77	72.02	75.55	71.05	72.15	70.40
CDT(Sun et al., 2019)	82.30	74.02	77.19	72.99	74.66	73.66
InterGCN(Liang et al., 2020)	82.23	74.01	77.86	74.32	-	-
R-GAT(Wang et al., 2020)	83.30	76.08	77.42	73.76	75.57	73.82
DGEDT(Tang et al., 2020)	83.90	75.10	76.80	72.30	74.80	73.40
DualGCN(Li et al., 2021a)	84.27	78.08	78.48	74.74	75.92	74.29
SSEGCN(Zhang et al., 2022)	84.72	77.51	**79.43**	**76.49**	76.51	75.32
MWGCN(Yu and Zhang, 2023)	82.56	74.58	76.36	72.28	72.86	70.73
DAGCN	**84.72**	**78.08**	78.96	75.07	**77.10**	**75.66**
LCF+BERT(Zeng et al., 2019)	87.14	81.74	82.45	79.59	77.31	75.78
R-GAT+BERT(Wang et al., 2020)	86.60	81.35	78.21	74.07	76.15	74.88
DGEDT+BERT(Tang et al., 2020)	86.30	80.00	79.80	75.60	77.90	75.40
BERT4GCN(Xiao et al., 2021)	84.75	77.11	77.49	73.01	74.73	73.76
T-GCN+BERT(Tian et al., 2021)	86.16	79.95	80.88	77.03	76.45	75.25
DualGCN+BERT(Li et al., 2021a)	87.13	81.16	81.80	78.10	77.40	76.02
SSEGCN+BERT(Zhang et al., 2022)	87.31	81.09	81.01	77.96	77.40	76.02
MWGCN+BERT(Yu and Zhang, 2023)	86.36	80.54	79.78	76.68	75.00	74.30
APARN(Ma et al., 2023)	87.76	82.44	81.96	79.10	**79.76**	**78.79**
DAGCN+BERT	**88.03**	**82.64**	**82.59**	**79.40**	78.73	78.01

3.6 Loss Function

To ensure that the features learned by AoGCN and SaGCN are distinct, we introduce the KL divergence to measure the difference between them. Suppose that the probability distributions of A^{ao} and A^{sa} are denoted as $P(X)$ and $Q(X)$, respectively, the KL divergence loss is calculated as follows:

$$\ell_{kl}(\theta) = \sum_{x \in X} P(x) log \frac{P(x)}{Q(x)} \tag{11}$$

Where θ represents all trainable parameters.

In addition, we also employ the standard cross-entropy loss function commonly used in ABSA, which can be defined as follows:

$$\ell_c(\theta) = - \sum_{(s,a) \in \mathcal{D}} \sum_{c \in \mathcal{C}} logp(a) \tag{12}$$

Where $\mathcal{D}$ contains all the sentence-aspect pairs and $\mathcal{C}$ is the set of sentiment polarities.

Then, we combine the KL divergence loss with the cross-entropy loss to obtain the final objective function:

$$\ell(\theta) = \ell_c(\theta) + \gamma \cdot \ell_{kl}(\theta) \tag{13}$$

Where γ ($\gamma < 0$) is a hyperparameter and $\ell(\theta)$ represents the objective function. The model parameters are optimized by minimizing the objective function.

4 Experiments

Statistics for the three experimental datasets and implementation details could be found in A.1 and A.2, respectively.

4.1 Main Results

As shown in Table 2, we compare the proposed model with previous works using evaluation metrics such as accuracy and macro F1-score. These baseline models are described in detail in A.3. The experimental results demonstrate that DAGCN outperforms all baseline models on the Restaurant dataset and are competitive to state-of-the-art (STOA) baseline models on Laptop and Twitter datasets. These results validate the effectiveness of the model architecture.

Compare with semantic models The experiment results of DAGCN highlight the importance of incorporating syntactic structures, compared with some attention-based methods (i.e., IAN, RAM, and TNet). Besides, DAGCN outperforms methods using SRD (i.e., LCF, MWGCN) on all datasets, no matter encoder is BiLSTM or BERT.

Table 3: Experimental results of ablation study.

Models	Restaurant		Laptop		Twitter	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
DAGCN	**84.72**	**78.08**	**78.96**	**75.07**	**77.10**	**75.66**
w/o KL divergence loss	83.47	76.79	76.74	72.93	73.41	71.50
w/o AF	82.84	75.19	76.58	72.27	74.00	72.85
w/o DWM	82.39	73.85	78.16	74.27	73.12	70.61
w/o DSW	82.84	75.07	76.11	73.16	74.30	72.44

Table 4: Case studies of our DAGCN model compared with other baselines.

Sentences	LCF	CDT	DAGCN	Target
it 's fast , light , and simple to *use*.	**P**	**P**	**P**	P
I complained to the waiter and then to the *manager*, but the intensity of rudeness from them just went up.	**N**	O	**N**	N
The food is so good and so popular that *waiting* can really be a nightmare.	P	**N**	**N**	N
The *mountain lion os* is not hard to figure out if you are familiar with microsoft windows.	N	N	**P**	P

It proves that DSW elevates the emphasis on the significance of opinion words and assists DAGCN in accurately identifying opinion words within a more confined local context.

Compare with syntactic models Our proposed model outperforms syntactic models (i.e., CDT, R-GAT and DGEDT) mostly in all datasets, because it can distinguish the contribution of context words in determining the aspect's sentiment polarity. Notably, when the encoder of syntactic models is BERT, our model achieves superior performance on Restaurant and Laptop datasets. Previous works relying solely on the dependency tree might introduce syntactic noise. They also require multiple aggregations to capture features of opinion words located further from the aspect, which can lead to overfitting. In our approach, we link opinion words to the aspect directly and enable more targeted aggregation in GCN.

Compare with SOTA model When encoder is BiLSTM, the proposed model performs worse than SSEGCN on Laptop dataset and the reason is that DAGCN's capability of capturing global semantic features is not as strong as SSEGCN's. More details could be found in Appendix A.4. We also notice that DAGCN performs worse compared to APARN on Twitter dataset, when encoder is BERT. The primary reason is that the AMR parser used in APARN has been trained on the dataset highly similar to Twitter dataset, making it more adapted to Twitter dataset. However, our model achieves comparable result on Twitter dataset. Meanwhile, the results on Restaurant and Laptop represent that formal language exhibits a more comprehensive and lucid syntactic structure. Overemphasizing semantic features might overlook the richness embedded within the syntactic information.

4.2 Ablation study

To validate the necessity of the proposed modules, we further conduct ablation experiments. As shown in Table 3, we first remove the KL divergence loss and utilize the loss function proposed in DualGCN as the objective function. The model's performance decreases, with a reduction in accuracy of 1.48%, 2.81%, and 4.79% on the Restaurant, Laptop, and Twitter datasets, respectively. This significant drop in performance demonstrates that the KL divergence loss effectively prevents AoGCN and SaGCN from learning redundant information and is also better than DualGCN's loss function. Furthermore, we remove AF and the model's performance is also compromised. Without AF, the model fails to aggregate global syntactic features. Next, by removing the DWM, we observe a significant decline in accuracy of 2.75% and 5.16% on the Restaurant and Twitter datasets, respectively. This further confirms the usefulness of pruning the dependency tree and that the utilization of DSW reveals the indispensability of local syntactic information. Finally, similar with AODT,

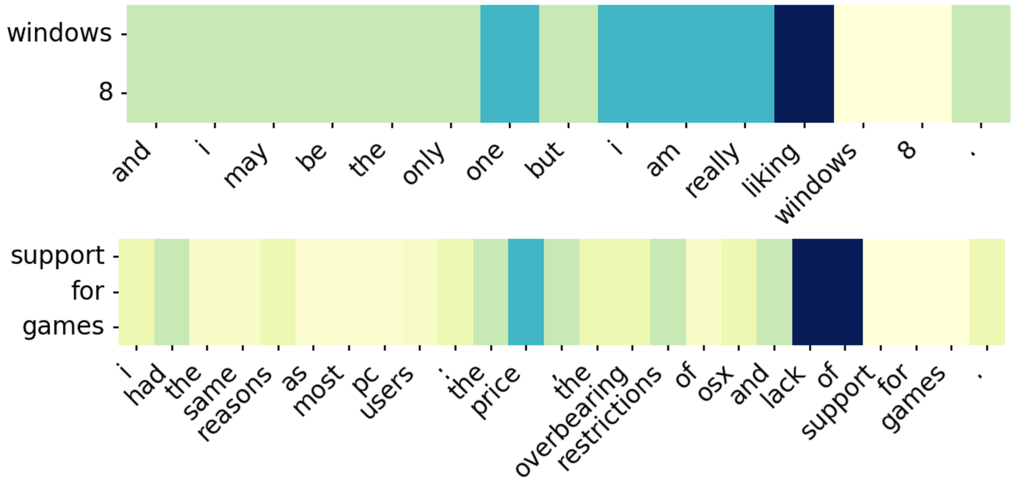


Figure 4: Two visualization examples of CSV in two cases.

we replace DSW with direct connection and set no links between context words and aspects when SD is longer than 4. We observe a decrease in the model's performance, which is attributed to the presence of non-opinion words within the local syntactic scope. Equal aggregation fails to highlight the importance of opinion words. In summary, each proposed module contributes significantly to the overall model, and their absence leads to a performance degradation.

4.3 Case study

To further analyze the performance of DAGCN, we conduct a detailed analysis on real examples. As shown in Table 4, we select LCF, CDT to compare their classification capabilities with DAGCN. In each example, the aspect is indicated in italics, and the notations P, N, and O represent positive, negative, and neutral sentiment, respectively. In the first example, the aspect is *"use"* and its corresponding opinion words are *"fast"*, *"light"*, and *"simple"*. These words have similar positive sentiment, making it unambiguous for LCF and CDT to quickly distinguish the sentiment polarity. In the second example, SRD between *"manager"* and the opinion word *"rudeness"* exhibits a small value, leading LCF to identify *"rudeness"* within the local context through attention mechanisms. However, the considerable SD between the aspect and the opinion word hampers information transmission. This illustrates the necessity of creating direct edges between them by removing irrelevant dependency relation and filtering words around the aspect. In the third example, due to the larger SRD value between *"nightmare"* and *"waiting"*, LCF prefers closer context words. While in the dependency tree, *"waiting"* has a strong syntactic relationship with *"nightmare"*, allowing CDT to make the correct judgment. This demonstrates that relying solely on semantics would be limited and syntax must be considered. Due to the accurate pruning of the dependency tree and the focus on local syntactic information, the proposed model precisely captures sentiment information corresponding to the opinion words. Moreover, in the final example, both LCF and CDT focus on *"hard"* while overlooking the negation's role in reversing sentiment. DAGCN, however, leverages AF to elevate the significance of negation words, achieving better performance.

4.4 Visualization

In Figure 4, we present visualizations of two illustrative instances to investigate the efficacy of the proposed model in discerning opinion words. In the initial case, the opinion word of *"liking"* to the pivotal aspect *"windows 8"* is exploited via DWM, elevating the salience of *"liking"*. Concurrently, AF intelligently mitigates the weights of other lexical entities (e.g., *"really"*) within the local scope, enabling the model to concentrate on pertinent opinion words. Moreover, the subsequent instance encompasses multiple aspects and opinion words. Without the confines of adhering to local syntactic distances, the model could inadvertently misattribute opinion words from distant aspects as classification criteria. However, by leveraging DWM and AF, we effectively obviate such external influences, enabling precise discernment of corresponding opinion words, thereby resulting in superior classification outcomes.

5 Conclusion

In this paper, we have presented a novel DAGCN model. With the statistical analysis of syntactic distance, there's a higher probability of opinion words appearing in the local context. Hence, we employ DSW to assign higher weights to words closer to the aspects in terms of syntactic distance, and it could also eliminate noise in the dependency tree. Furthermore, we construct DWM to store DSW and update DWM by AF to enhance the accuracy of identifying opinion words and accommodate scenarios where opinion words are distant from the aspects. Finally, inspired by previous work, we define a SaGCN to deal with some reviews with unstructured syntax and KL divergence is integrated into the loss function to guarantee distinct learning for AoGCN and SaGCN. Compared with other baselines, DAGCN achieves superior performance on public datasets, which demonstrates the effectiveness of the proposed architecture.

6 Limitations

Firstly, DAGCN could not outperform APARN on the Twitter dataset, when using BERT as the encoder. This might be attributed to the fact that CoreNLP excels at parsing syntactically structured sentences but may not perform as well on informal expressions as AMR. In future work, we aim to leverage the advantages of CoreNLP and AMR to enhance ABSA task.

Secondly, DAGCN employs additional hyperparameters (α, β and γ) that require extensive experiments to optimize the model. This process demands significant time and computational resources. Therefore, transitioning from manual hyperparameter selection to adaptive parameter tuning is highly justified.

Lastly, DAGCN primarily addresses the core problem of sentiment classification in this paper and has not been adapted for end-to-end ABSA and ASTE tasks. We plan to investigate DAGCN's generalization capabilities for complex ABSA tasks in our future work.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grant 62372300, Grant 62302306, and Grant 62201350, and in part by the National Key Research and Development Program of China under Grant No.2022YFB4501704, and in part by the Research Base of Online Education for Shanghai Middle and Primary Schools.

References

Bingliang Chen, Guojun Lu, Yun Xue, and Qianhua Cai. 2021. A novel bi-branch graph convolutional neural network for aspect level sentiment classification. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Chenhua Chen, Zhiyang Teng, Zhongqing Wang, and Yue Zhang. 2022. Discrete opinion tree induction for aspect-based sentiment analysis. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2051–2064, Dublin, Ireland. Association for Computational Linguistics.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 452–461, Copenhagen, Denmark. Association for Computational Linguistics.

Dong Deng, Liping Jing, Jian Yu, and Shaolong Sun. 2019. Sparse self-attention lstm for sentiment lexicon construction. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(11):1777–1790.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

Zhifang Fan, Zhen Wu, Xinyu Dai, Shujian Huang, and Jiajun Chen. 2019. Target-oriented opinion words extraction with target-fused neural sequence labeling. 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)*, pages 2509–2518.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018. Effective attention modeling for aspect-level sentiment classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1121–1131, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. 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)*, pages 6280–6285.

Tingyao Jiang, Zilong Wang, Ming Yang, and Cheng Li. 2023. Aspect-based sentiment analysis with dependency relation weighted graph attention. *Information*, 14(3):185.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Suin Kim, Jianwen Zhang, Zheng Chen, Alice Oh, and Shixia Liu. 2013. A hierarchical aspect-sentiment model for online reviews. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 526–533.

Ruifan Li, Hao Chen, Fangxiang Feng, Zhanyu Ma, Xiaojie Wang, and Eduard Hovy. 2021a. 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)*, pages 6319–6329, Online. Association for Computational Linguistics.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018. Transformation networks for target-oriented sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 946–956, Melbourne, Australia. Association for Computational Linguistics.

Yuanlin Li, Xiao Sun, and Meng Wang. 2021b. Embedding extra knowledge and a dependency tree based on a graph attention network for aspect-based sentiment analysis. In *2021 international joint conference on neural networks (IJCNN)*, pages 1–8. IEEE.

Bin Liang, Rongdi Yin, Lin Gui, Jiachen Du, and Ruifeng Xu. 2020. Jointly learning aspect-focused and inter-aspect relations with graph convolutional networks for aspect sentiment analysis. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 150–161, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Qiao Liu, Haibin Zhang, Yifu Zeng, Ziqi Huang, and Zufeng Wu. 2018. Content attention model for aspect based sentiment analysis. In *Proceedings of the 2018 world wide web conference*, pages 1023–1032.

Yifan Liu, Luwei Xiao, Yue-Cai Huang, Yun Xue, Xiaohui Hu, Haoliang Zhao, and Ying Li. 2022. Aspect-level sentiment analysis with local semantic and global syntactic features integration. *INTERNATIONAL JOURNAL OF PATTERN RECOGNITION AND ARTIFICIAL INTELLIGENCE*, 36(03).

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017a. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4068–4074.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017b. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, page 4068–4074. AAAI Press.

Fukun Ma, Xuming Hu, Aiwei Liu, Yawen Yang, Shuang Li, Philip S. Yu, and Lijie Wen. 2023. AMR-based network for aspect-based sentiment analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 322–337, Toronto, Canada. Association for Computational Linguistics.

Yukun Ma, Haiyun Peng, Tahir Khan, Erik Cambria, and Amir Hussain. 2018. Sentic lstm: a hybrid network for targeted aspect-based sentiment analysis. *Cognitive Computation*, 10:639–650.

Refuoe Mokhosi, ZhiGuang Qin, Qiao Liu, and Casper Shikali. 2019. Incorporating word significance into aspect-level sentiment analysis. *APPLIED SCIENCES-BASEL*, 9(17).

Huy Thanh Nguyen and Minh Le Nguyen. 2018. Effective attention networks for aspect-level sentiment classification. In *2018 10th International Conference on Knowledge and Systems Engineering (KSE)*, pages 25–30.

Thien Hai Nguyen and Kiyoaki Shirai. 2015. PhraseRNN: Phrase recursive neural network for aspect-based sentiment analysis. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2509–2514, Lisbon, Portugal. Association for Computational Linguistics.

Kai Sun, Richong Zhang, Samuel Mensah, Yongyi Mao, and Xudong Liu. 2019. 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)*, pages 5679–5688, Hong Kong, China. Association for Computational Linguistics.

Hao Tang, Donghong Ji, Chenliang Li, and Qiji Zhou. 2020. Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6578–6588, Online. Association for Computational Linguistics.

Siyu Tang, Heyan Chai, Ziyi Yao, Ye Ding, Cuiyun Gao, Binxing Fang, and Qing Liao. 2022. Affective knowledge enhanced multiple-graph fusion networks for aspect-based sentiment analysis. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5352–5362.

Mike Thelwall and Kevan Buckley. 2013. Topic-based sentiment analysis for the social web: The role of mood and issue-related words. *Journal of the American Society for Information Science and Technology*, 64(8):1608–1617.

Yuanhe Tian, Guimin Chen, and Yan Song. 2021. 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, pages 2910–2922, Online. Association for Computational Linguistics.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3229–3238, Online. Association for Computational Linguistics.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016a. Recursive neural conditional random fields for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 616–626.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016b. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615.

Zeguan Xiao, Jiarun Wu, Qingliang Chen, and Congjian Deng. 2021. 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*, pages 9193–9200, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zehao Yan, Shiguan Pang, and Yun Xue. 2021. Semantic enhanced dual-channel graph communication network for aspect-based sentiment analysis. In *Natural Language Processing and Chinese Computing: 10th CCF International Conference, NLPCC 2021, Qingdao, China, October 13–17, 2021, Proceedings, Part I 10*, pages 531–543. Springer.

Bengong Yu and Shuwen Zhang. 2023. A novel weight-oriented graph convolutional network for aspect-based sentiment analysis. *The Journal of Supercomputing*, 79(1):947–972.

Biqing Zeng, Heng Yang, Ruyang Xu, Wu Zhou, and Xuli Han. 2019. Lcf: A local context focus mechanism for aspect-based sentiment classification. *Applied Sciences*, 9(16).

Chen Zhang, Qiuchi Li, and Dawei Song. 2019. 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)*, pages 4568–4578, Hong Kong, China. Association for Computational Linguistics.

Zheng Zhang, Zili Zhou, and Yanna Wang. 2022. 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*, pages 4916–4925, Seattle, United States. Association for Computational Linguistics.

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, Hua Jin, and Dacheng Tao. 2023. Knowledge graph augmented network towards multiview representation learning for aspect-based sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–14.

Yuxiang Zhou, Lejian Liao, Yang Gao, Zhanming Jie, and Wei Lu. 2021. To be closer: Learning to link up aspects with opinions. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3899–3909, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Appendix

A.1 Datasets

We conduct experiments on three public benchmark datasets for ABSA. Restaurant and Laptop reviews datasets are from SemEval 2014 Task 4 and Twitter dataset consists of tweets. In the Twitter dataset, we exclude tweets with the "conflict" label. All datasets contain data with three sentiment polarities: positive, neutral, and negative. The aspect terms and sentiment polarities have been annotated in the datasets. The statistics of the three datasets are shown in Table 5. In this paper, we follow the Creative Commons Attribution 4.0 International Licence of the datasets.

A.2 Implementation Details

We utilize Stanford's CoreNLP[2] as the dependency parser in our approach. We initialize word embeddings using 300-dimensional GloVe[3] vectors as a lookup table. In the encoder, the dimensions of the hidden states for BiLSTM and BERT are set to 50 and 768, respectively, with a dropout rate of 0.7. We use the bert-base-uncased[4] version of BERT. When encoder is BiLSTM, the model is trained for 50 epochs and takes approximately 26s to train one epoch on a single RTX 3090 GPU with the batch size of 16. When encoder is BERT, epochs are set to 15 and DAGCN takes approximately 96s to train one epoch on a single RTX 3090 GPU with batch size of 16. The total parameter sizes of DAGCN are about 1.2M and 113M, respectively.

[2]https://stanfordnlp.github.io/CoreNLP/

[3]https://nlp.stanford.edu/projects/glove/

[4]https://github.com/huggingface/transformers

Table 5: Statistics for the three experimental datasets.

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

The AoGCN and SaGCN have a layer depth of 1 and a dropout rate of 0.1. We optimize the parameters using the Adam optimizer with a learning rate of 0.002. The three hyperparameters, α, β, and γ, are set to (-0.7, 0.9, -0.3), (-0.7, 0.3, -0.8), and (-0.2, 0.6, -0.2) for three datasets respectively.

A.3 Baseline Models

To thoroughly evaluate the effectiveness of our proposed model, we compare DAGCN against state-of-the-art baselines, including:

1. **IAN** (Ma et al., 2017b) proposes a new approach for ABSA by separately modeling the targets and contexts using interactive attention networks.

2. **RAM** (Chen et al., 2017) integrates a recurrent neural network with a weighted-memory mechanism to capture sentiment features.

3. **TNet** (Li et al., 2018) combines a BiLSTM layer with a CNN layer to extract salient features from transformed word representations.

4. **LCF** (Zeng et al., 2019) introduces a new idea that the local context of aspects contains significant information and SRD is proposed to pay more attention in local scope.

5. **ASGCN** (Zhang et al., 2019) first employs a GCN to learn aspect representations in aspect based sentiment analysis task.

6. **CDT** (Sun et al., 2019) uses a BiLSTM for learning sentence features and a GCN is applied to the dependency tree to enhance the embeddings.

7. **InterGCN** (Liang et al., 2020) constructs a heterogeneous graph for each instance by leveraging aspect-focused and inter-aspect contextual dependencies

8. **R-GAT** (Wang et al., 2020) encodes syntax information through a aspect-oriented dependency tree structure and introduces dependency relation into convolution.

9. **DGEDT** (Tang et al., 2020) proposes a dependency graph enhanced dual-transformer network that utilizes a dual-transformer structure to mutually reinforce the flat and graph-based representations.

10. **DualGCN** (Li et al., 2021a) utilizes two GCNs to learn syntactic information and semantic information, respectively.

11. **SSEGCN** (Zhang et al., 2022) proposes an aspect-aware attention mechanism with self-attention to learn aspect-related and global semantics of a sentence and then combines them with syntactic information.

12. **MWGCN** (Yu and Zhang, 2023) generates a local context weighted adjacency graph based on SRD and proposes another weighting method to retain global semantics.

13. **LCF+BERT** (Zeng et al., 2019) is the lcf model whose encoder is replaced by a pre-trained BERT.

14. **R-GAT+BERT** (Wang et al., 2020) is the R-GAT model whose encoder is replaced by a pre-trained BERT.

15. **DGEDT+BERT** (Tang et al., 2020) is the DGEDT model whose encoder is replaced by a pre-trained BERT.

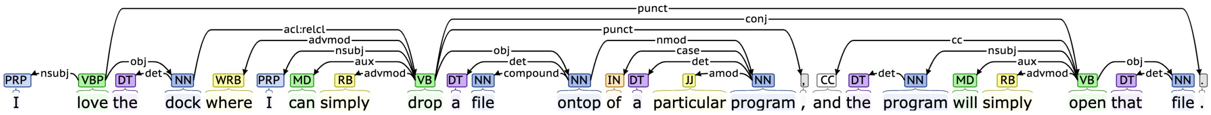
16. **BERT4GCN** (Xiao et al., 2021) integrates the contextual features output from BERT and the syntactic knowledge from dependency graphs.

17. **T-GCN+BERT** (Tian et al., 2021) utilizes attention and layer ensemble to explicitly consider dependency types in the graph.

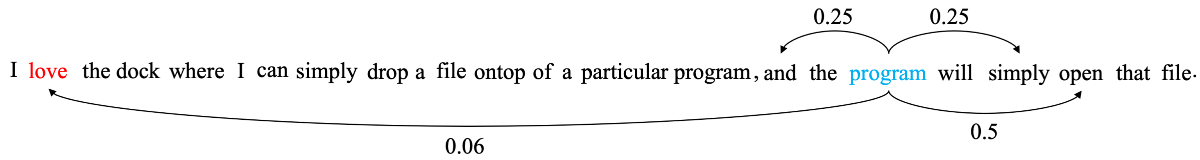
18. **DualGCN+BERT** (Li et al., 2021a) is the DualGCN model whose encoder is replaced by a pre-trained BERT.

19. **SSEGCN+BERT** (Zhang et al., 2022) is the SSEGCN model whose encoder is replaced by a pre-trained BERT.

20. **MWGCN+BERT** (Yu and Zhang, 2023) is the MWGCN model whose encoder is replaced by a pre-trained BERT.



(a) The dependency tree



(b) A part of AODT with DSW

Figure 5: Error Analysis on Laptop Dataset.

21. **APARN** (Ma et al., 2023) employs semantic structure called Abstract Meaning Representation to have abundant semantic representation and integrates it with attention mechanism to improve sentence features.

A.4 Error Analysis on Laptop Dataset

In this section, we investigate the reason why the proposed model performs worse than SSEGCN on Laptop dataset. An instance is selected for analysis, which is "I love the dock where I can simply drop a file ontop of a particular program, and the **program** will simply open that file.". The second *"program"* is the aspect and its corresponding sentiment polarity is positive. Figure 5a shows the dependency tree of the instance. In order to make the figure concise and aesthetically pleasing, Figure 5b shows just a part of AODT with DSW.

From the human perspective, *"dock"* and the aspect are in a parallel relationship and their sentiment polarity should be the same. SSEGCN makes the correct classification because it has the ability to capture hierarchical semantic features and obtain long-distance semantic commonalities. However, the proposed model determines this aspect to be neutral. The reason perhaps is that *“love”* has low DSW (DSW=0.06) and AF fails to grasp very long-distance semantic relationship. Therefore, DAGCN tends to focus on syntactic information and have a relatively weaker grasp of global semantics. In future work, we will continue to enhance the model’s long-distance semantic understanding capabilities.

A.5 MAMS Results

To further verify the effectiveness and robustness, we conduct another experiment on MAMS dataset (Jiang et al., 2019). Followed by Li et al. (2021a), we remove instances with "conflict" label. Table 6 shows the statistics for MAMS dataset. Followed by Ma et al. (2023), we compare the performance between DAGCN and other baseline models when encoder is BERT.

Table 6: Statistics for MAMS dataset.

Dataset	Split	Positive	Neutral	Negative
MAMS	Train	3380	5042	2764
	Dev	403	604	325
	Test	400	607	329

Table 7: Experimental results comparison on MAMS dataset.

Models	MAMS	
	Accuracy	Macro-F1
BERT(Kenton and Toutanova, 2019)	80.11	80.34
T-GCN(Tian et al., 2021)	83.38	82.77
dotGCN(Chen et al., 2022)	84.95	84.44
APARN(Ma et al., 2023)	**85.59**	**85.06**
DAGCN	85.25	84.87

As shown in Table 7, DAGCN outperforms most of baseline models and achieves comparable results compared to APARN. Note that dotGCN similarly prunes the dependency tree and introduces multiple additional loss functions. However, in comparison to dotGCN, our model’s tree pruning is more intuitive, and we have only introduced one additional loss function. From the experimental results, it is evident that our model surpasses dotGCN, thereby affirming the effectiveness and robustness of our proposed approach.

A.6 Effect of the dependency parser

In order to validate generalization, we conduct an study based on the proposed method using another dependency parser: Biaffine Parser (Dozat and Manning, 2016). Table 8 shows the performance of dependency parsers when encoder is

Table 8: Experimental results comparison on different dependency parser.

Dependency Parser	Restaurant		Laptop		Twitter	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
Biaffine Parser	83.47	76.57	78.01	74.71	75.92	74.87
CoreNLP	**84.72**	**78.08**	**78.96**	**75.07**	**77.10**	**75.66**

GloVe and the result of CoreNLP is same with the result of DAGCN in Table 2. We can find easily that CoreNLP performs better than Biaffine Parser. Besides, DAGCN with Biaffine Parser also surpasses most of baseline models introduced in Table 2. It demonstrates that DAGCN's performance would not change dramatically when using different dependency parsers and further affirms that our model has stronger generalization.

A.7 Effect of the DAGCN Layer Number

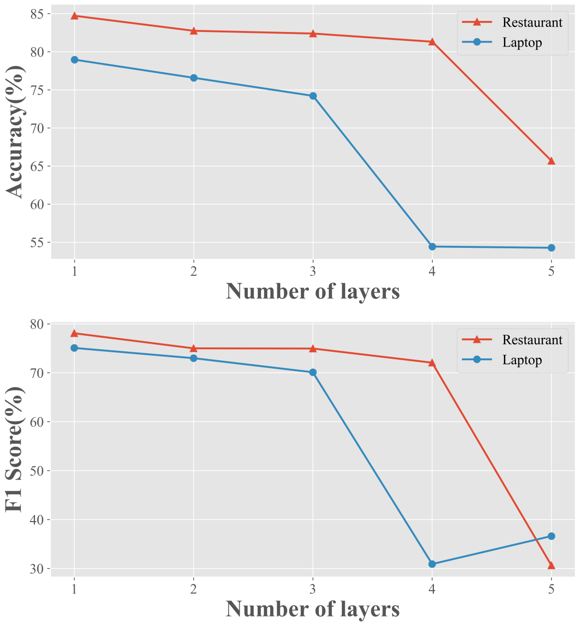


Figure 6: Effect of the number of DAGCN layers.

In this section, we investigate the impact of DAGCN layer number on the performance. Figure 6 illustrates the changes in accuracy and macro-F1 scores on the Restaurant and Laptop datasets as the layer number varies from 1 to 5. From the results, we can observe that the model performs optimally when the number of GCN layers is 1. As the number of layers increases, particularly when it reaches 5, the performance diminishes. This is attributed to the direct connection between the aspects and context words through the construction of DWM. With fewer layers, the model avoids excessive aggregation operations, whereas a higher number of layers can lead to overfitting.