

Dynamic and Multi-Channel Graph Convolutional Network for Aspect-Based Sentiment Analysis

Shiguan Pang¹, Yun Xue*, Zehao Yan², Weihao Huang², Jinhui Feng²

School of Physics and Telecommunication Engineering

South China Normal University Guangzhou, China

{psg-nlp, xueyun}@m.scnu.edu.cn

Abstract

Aspect-based sentiment analysis aims to judge the polarity of the given aspect word in reviews. Most recent methods adopt syntax-based Graph Neural Networks to extract the syntactic information from the dependency graph, thinking that would be beneficial for establishing relations between aspect and opinion words. However, these methods may ignore that some sentences have no remarkable syntactic structure, which causes the opposite judgement in sentiment analysis. In this paper, we figure out this problem by means of optimally fusing syntactic information, semantic information and their combinations simultaneously. Firstly, syntactic graphs and semantic graphs are generated by dependency tree and multi-head self-attention respectively. Then we propose a Dynamic and Multi-channel Graph Convolutional Network (DM-GCN) to learn the correlated information from the generated graphs effectively. Our extensive experiments on SemEval 2014 and Twitter datasets confirm that DM-GCN fuses syntactic, semantic and their combinations optimally and outperforms all state-of-the-art alternatives with a large margin.

1 Introduction

Aspect-Based Sentiment Analysis (ABSA) is a fine-grained task of sentiment analysis. Distinguishing from document-based and sentence-based sentiment analysis, aspect-based sentiment analysis aims to determine the sentiment polarities of specific aspect words in given texts. As an example of ABSA, in the sentence '*The food is good but the service is bad*', the sentiment polarities of given aspect words *food* and *service* are recognized as *Positive* and *Negative*. As such a more detailed analysis of multiple aspect sentiment gives rise to provide targeted insights into reviews.

In general, there are several steps when performing aspect-based sentiment analysis, such as obtain word embeddings, encode the syntactic information and extract semantic information, among which mining the most relevant opinion word plays the pivot role. Benefited from applying attention mechanisms to connect aspect words with their opinion words, a few studies (Wang et al., 2016b; Li et al., 2017; Ma et al., 2017; Fan et al., 2018) have reported attractive results. However, limited by the co-occurrence frequency or long-range word dependencies, attention mechanism may assign wrong value for irrelevant words. As an example, '*the stuff should be more friendly*', due to the high co-occurrence frequency of '*stuff*' and '*friendly*', attention mechanisms may regard '*friendly*' as the opinion word for '*staff*', which result in an opposite sentiment judgement completely.

To address those limitations, a slice of studies encode the syntactic structure as crucial information to establish the connections between aspects and their opinion words. A handcrafted syntactic rule (Liu et al., 2013) has been encoded into the model as an early essay. However, it is over-relied upon the quality of the rule. Then dependency trees are regarded as another form of syntactic rule, (Dong et al., 2014; Nguyen and Shirai, 2015; Wang et al., 2016a) have encoded dependency tree by a Recursive Neural Network (RNN). Recently, the rapid development of graph neural networks has attracted a surge of interest, due to the great capacity of learning structure representation, a class of graph neural networks is designed to extract syntactic information from dependency tree (Zhang et al., 2019a; Sun et al., 2019; Huang and Carley, 2019; Wang et al., 2020b; Zheng et al., 2020). Although these approaches make quite a few improvements compared with those attention-based models, the shortcomings should not be ignored. First and foremost, sentences have different sensitivities to syn-

tactic information and semantic information. In particular, those sentences that have inconspicuous syntactic structure, are low sensitive to syntactic information, which means that syntactic information may fail to help the model to determine the sentiment polarity of the sentences in some cases. Given as an example, ‘Charlie sheen your fucking annoying’, a sentence in twitter dataset, prefers to colloquial expression, without explicit syntactic structure, which makes those extracted structure information become noise. Besides, not all the information on the dependency tree is valid for our task, the noise would be encoded by the GNNs-based model, then the secondary noise will be caused if attention mechanism is used on this basis (Zhang et al., 2019a). To make matter worse, GCNs has been proved that fail to optimally integrate node features (semantical information) and topological structures (syntactic structure) in a dependency graph with rich information (Wang et al., 2020b), indicating that GCN may arduous to learn some deep correlation between topological structures (syntactic structures) and node features (semantical information).

In reality, syntax is complementary to semantics. Motivated by this fundamental assumption, our Dynamic and Multi-Channels Graph Convolutional Network (DM-GCN) is designed to get over the weakness mentioned above. Firstly, on purpose to exploit the information in semantic space adequately, the semantic graphs are generated and updated dynamically by a multi-head self-attention mechanism in the graph convolutional network. Secondly, the dependency trees of sentences are transformed into syntactic graphs, which is the same as the model mentioned above (Sun et al., 2019; Zhang et al., 2019a). With the semantic graphs and syntactic graphs, two specific graph convolutional networks are applied to extract two specific information from semantic and syntactic space. Furthermore, considering that the common information of semantic and syntactic space is as significant as the specific information, a parameter sharing graph convolution module is designed to extract the common information. Finally, a trainable parameter matrix is utilized to adaptively learn the importance of different embeddings and fuse them optimally. Extensive experiments are carried out on SemEval2014 and Twitter datasets, and experimental results demonstrate that DM-GCN achieves predominant performance compared with baseline

methods.

The contributions of this paper can be summarized as follows:

- We have come up with a kind of dynamic semantic graph, generated and updated by the multi-head self-attention mechanism, which focuses on extract the most relevant information in the semantic space.
- We propose a novel Dynamic and Multi-channel GCN(DM-GCN) to adaptively learn and fuse according to the characteristics of sentences.
- Extensive results testify the significance of leveraging syntactical, semantical, and their combinational information properly, and demonstrate the availability of our proposed model in extracting and fusing them in sentiment classification.

The rest of the paper is organized as follows. In Section 2 we briefly review the related work of aspect-based sentiment analysis. In Section 3 we develop our proposed model DM-GCN. The evaluation of our model on benchmark datasets and the experimental result analysis are in Section 4. Finally, concluding remarks are presented in Section 5.

2 Related Work

With the appealing development of aspect-based sentiment analysis, numbers of research can be divided into three categories approximately: Attention-based neural models, Syntactic rules neural network, and Syntactic-based graph neural network. Our work is based on a plentiful line of those recent efforts.

Attention-Based Neural Networks: Originated in computer version, attention-based models have led to promising progress in sentiment analysis as well, which are committed to building up connection between aspect word and opinion word precisely. Among them, (Wang et al., 2016b) proposed an attention-based LSTM to obtain significant information for given aspects. (Chen et al., 2017)utilized a multi-layer attention network to address long-distance dependency problem between aspect and opinion words. With transformer being proposed, the pre-trained language model make

impressive achievement in NLP including aspect-based sentiment analysis (Xu et al., 2019; Sun et al., 2019; Vaswani et al., 2017).

Syntactic-Based Recurrent Neural Networks:

Due to the lack of structural information, attention-based methods may assign the attention weights mistakenly. Some research tries to address this issue by encoding syntactic rules in word representation. (Dong et al., 2014) proposed AdaRNN, which can propagate the sentiment information via syntactic dependency tree. (He et al., 2018a) proposed an attention mechanism that incorporates syntactic information. (Zhang et al., 2019b) tried to utilize the distance between aspect word and context on syntactic dependency tree as auxiliary weight.

Syntactic-Based Graph Neural Networks:

More recently, with the rapid development of graph neural networks (Kipf and Welling, 2019; Velickovic et al., 2018), the combination of syntactic dependency tree and graph neural networks have shown gratifying results in ABSA. (Zhang et al., 2019a; Sun et al., 2019) transformed syntactic dependency tree into graph structure and utilized GCN to extract syntactic information. (Tang et al., 2020) proposed Bidirectional-GCN to encode the directional information of syntactic dependency tree. (Wang et al., 2020a) proposed a new GAT model to encode the dependency relations. However, whether the correlated information can be extracted according to the characteristics of the sentence from GNNs adaptively remains indistinct.

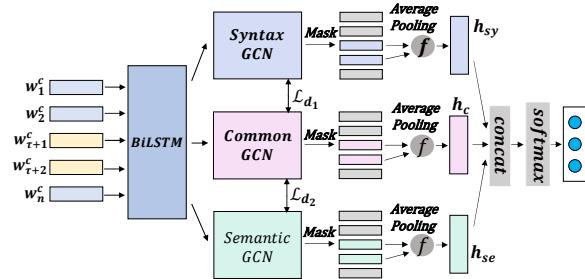


Figure 1: Model Architecture

3 Methodology

In this section, we elaborate on the details of our proposed model. The overall framework of DM-GCN is shown in Figure 1. Inspired by AM-GCN (Wang et al., 2020b), our key idea is that the proposed model should be available to aggregate syn-

tactical features, semantical features and their combination features becomingly to address the issues mentioned before. For this purpose, firstly, semantic graphs are constructed by the multi-head self-attention mechanism. Secondly, syntactic graphs are transformed from the dependency tree of sentences. Then two specific convolution modules and a common convolution module are used to extract corresponding information respectively. Finally, a trainable parameter is used to fuse extracted information more suitably for our task. The components of DM-GCN will be illustrated in the rest of the section.

3.1 Embedding and Bidirectional LSTM

An n-word sentence $x = \{w_1^c, w_2^c, \dots, w_{\tau+1}^c, \dots, w_{\tau+m}^c, \dots, w_n^c\}$ with the aspect $\{w_{\tau+1}^c, \dots, w_{\tau+m}^c\}$ is given, each word in the sentence will be embedded into a low-dimensional vector by looking up in a pre-trained word embedding matrix $E \in R^{|V| \times d_e}$, where $|V|$ is the lexicon size and d_e is the dimension of word embedding (Bengio et al., 2003). With the sentence embeddings, a bidirectional LSTM is constructed to learn the hidden representation of the given sentence x , which can be conveyed as $\vec{H}^c = \{\vec{h}_1^c, \vec{h}_2^c, \dots, \vec{h}_{\tau+1}^c, \dots, \vec{h}_{\tau+m}^c, \vec{h}_{n-1}^c, \vec{h}_n^c\}$ in the forward direction and $\tilde{H}^c = \{\tilde{h}_1^c, \tilde{h}_2^c, \dots, \tilde{h}_{\tau+1}^c, \dots, \tilde{h}_{\tau+m}^c, \tilde{h}_{n-1}^c, \tilde{h}_n^c\}$ in the backward direction. Concatenating the forward and backward representation, $H^c = [\vec{H}^c, \tilde{H}^c] \in R^{n \times d_{lstm}}$ is the final representation encoded by Bidirectional LSTM, which contains the contextual information between aspect words and opinion words.

3.2 Syntactic Graph Convolution Module

As previous described (Sun et al., 2019; Zhang et al., 2019a), we transform the dependency tree into the graph structure $G_{sy} = (A_{sy}, H^c)$, A_{sy} is the adjacency matrix of the graph and H^c is the feature matrix. Then the graph convolutional network is applied to extract the syntactic information, which is formulated as below:

$$GCN(\tilde{A}, H^{(l)}, W^{(l+1)}) = ReLU(\tilde{A}H^{(l)}W^{(l+1)}) \quad (1)$$

$$H_{sy}^{(0)} = H^c \quad (2)$$

$$H_{sy,in}^{(l)} = [H_{sy}^{(0)}; \dots; H_{sy}^{(l)}] \quad (3)$$

$$H_{sy}^{(l+1)} = GCN(\tilde{A}_{sy}, H_{sy,in}^{(l)}, W_{sy}^{(l+1)}) \quad (4)$$

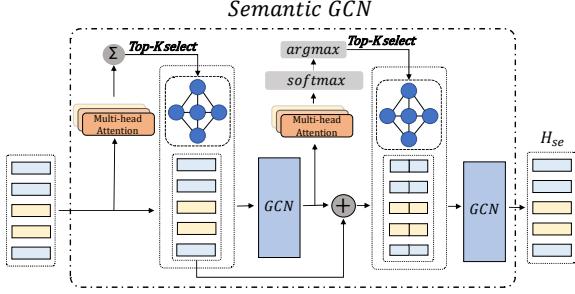


Figure 2: Architecture of Semantic Graph Convolution Module

Where $\tilde{A}_{sy} = \tilde{D}^{-\frac{1}{2}} (A_{sy} + I_f) \tilde{D}^{-\frac{1}{2}}$ is the adjacency matrix with self-loop (Kipf and Welling, 2019), $H_{sy,in}^{(0)} = H^c \in R^{n \times d_{lstm}}$ is the words representation encoded by Bi-LSTM, we take it as the input of the first GCN layer. When $l \in [1, L - 1]$ the input $H_{sy,in}^{(l)}$ is generated by the concatenation of the output produced in layers $0, \dots, l - 1$, $W_{sy}^{(l+1)} \in R^{(d_{lstm} + l * d_{gcn}) \times d_{lstm}}$ is the learnable matrix of the l -th GCN layer, d_{lstm} is the dimension of the hidden representation learned by bidirectional LSTM, d_{gcn} is the dimension of the GCN-layer output and d_{head} is the dimension of each attention head. With l -step convolutional operation, each node can iteratively aggregate the information from its one-hop neighbors and update its representation. Syntactic information is integrated into the final representation $H_{sy}^{(L)}$ successfully by Syntactic Graph Convolutional Module.

3.3 Semantic Graph Convolution Module

As we discuss in Section 1, a few short sentences have vague syntactic structure, if the model extracts the syntactic information stiffly, may trigger the opposite effect. On the other, the original dependency tree can be seen as a form of hard attention mechanism (Xu et al., 2015), which may lead to the information omission when aspects are far away from its opinion words or two words are supposed to be connected but they don't on their dependency tree. Inspired by (Huang and Carley, 2019), we develop Semantic Graph Convolutional Module to distinguish the correlated context words directly.

Specifically, the semantic graph convolution module is divided into initialization and update respectively. we take $H_{se,in}^{(0)}$ as the input of the initialization, the multi-head self-attention will be utilized to construct k attention score matrices based on $H_{se,in}^{(0)}$. On purpose to enhance the robustness of the model, these k matrices will be summed up

intuitively. Then a $top-k$ selection is applied on the summed-up matrix, which means that the $top-k$ significant context words are preserved. More specially, the adjacency matrix is supposed to keep symmetry after $top-k$ selection. In this way, we have initialized the adjacency matrix $A_{se}^{(0)}$ and a GCN layer is applied to extract the semantic information preliminarily. The initialization is formulated as:

$$H_{se,in}^{(0)} = H^c \quad (5)$$

$$A_{se,i}^{(0)} = \frac{(H_{se,in}^{(0)} W_{se,k}^{(0)}) (H_{se,in}^{(0)} W_{se,q}^{(0)})^T}{\sqrt{d_{head}}} \quad (6)$$

$$d_{head} = \frac{d_{lstm}}{K} \quad (7)$$

Specially, when $l = 0$:

$$A_{se}^{(0)} = top-k(\sum_{i=0}^K A_{se,i}^{(0)}) \quad (8)$$

$$H_{se}^{(1)} = GCN(\tilde{A}_{se}^{(0)}, H_{se,in}^{(0)}, W_{se}^{(1)}) \quad (9)$$

where $H_{se,in}^{(0)} \in R^{n \times d_{lstm}}$ is the words representation encoded by Bi-LSTM, K is the number of heads, $A_{se,i}^{(0)}$ is the i -th adjacency matrix, $W_{se,k}^{(0)}, W_{se,q}^{(0)} \in R^{d_{lstm} \times d_{head}}$ are trainable weight matrices, $W_{se}^{(1)} \in R^{d_{lstm} \times d_{gcn}}$ is the parameter matrix of the GCN layer.

With the initialized semantic graph $A_{se}^{(0)}$, the process of update is formulated as :

$$H_{se,in}^{(l)} = [H_{se}^{(0)}; \dots; H_{se}^{(l)}] \quad (10)$$

$$A_{se,i}^{(l)} = \frac{(H_{se,in}^{(l)} W_{se,k}^{(l)}) (H_{se,in}^{(l)} W_{se,q}^{(l)})^T}{\sqrt{d_{head}}} \quad (11)$$

$$A_{se}^{(l)} = argmax[softmax(A_{se,1}^{(l)}, \dots, A_{se,K}^{(l)})] \quad (12)$$

$$A_{se}^{(l)} = top-k(A_{se}^{(l)}) \quad (13)$$

$$H_{se}^{(l+1)} = GCN(\tilde{A}_{se}^{(l)}, H_{se,in}^{(l)}, W_{se}^{(l+1)}) \quad (14)$$

Put slightly different, the input $H_{se,in}^{(l)}$ is generated by the concatenation of the output produced

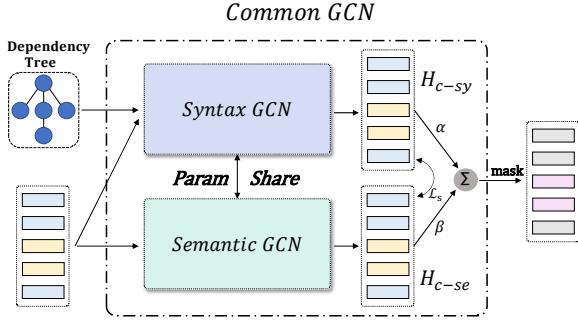


Figure 3: Architecture of Common Graph Convolution Module

in layers $0, \dots, l-1$, which will be delivered into the multi-head self-attention to generate k attention score matrices just as initialization does. Then we figure out the matrix with maximum probability by *softmax* function, which can be regarded as the most relevant attention matrix of the semantic graph for the current sentence. After *top-k* selection, a GCN layer is utilized to extract the deep-seated semantic information. We denote the last layer output representation as $H_{se}^{(L)}$, which contains the most relevant semantic information of aspects and its opinion word.

3.4 Common Convolution Module

According to (Liina and Pylkknen, 2019), syntax and semantic space are not totally separating, as the syntactic structure of sentence changes, so does the semantic, which gives reason to believe that extracting the common information shared by syntactic and semantic space is beneficial to comprehend the sentence better. Motivated by (Wang et al., 2020b), a common graph convolutional module (Figure 3) with parameter sharing strategy is applied to obtain the information shared in two spaces.

For simplicity, we represent the Syntactic Graph Convolution Module and the Semantic Graph Convolution Module as:

$$H_{sy}^{(L)} = SYGCN(A_{sy}, H^c, W_{sy}) \quad (15)$$

$$H_{se}^{(L)} = SEGCGN(A_{se}, H^c, W_{se}) \quad (16)$$

the common syntactic representation $H_{c-sy}^{(L)}$ is extracted by common graph convolutional module from syntactic graph $G_{sy} = (A_{sy}, H^c)$ as follow

$$H_{c-sy}^{(L)} = SYGCN(A_{sy}, H^c, W_c) \quad (17)$$

Where $W_c = W_c^{(l+1)} \in R^{(d_{lstm} + l * d_{gcn}) \times d_{gcn}}$ is the l -th layer learnable weight matrix of Common-

GCN, $H_{c-sy}^{(L)}$ is the final common syntactic representation.

On purpose to extract the shared information, when applying *Common-GCN* to learn the node representation from semantic graph $G_{se} = (A_{se}, H^c)$, we share the same learnable weight matrix $W_c^{(l)}$ for each layer of *Common-GCN*.

Simply, the calculation is similar to section 3.3:

$$H_{c-se}^{(L)} = SEGCGN(A_{se}, H^c, W_c) \quad (18)$$

Finally, we combine these two representations $H_{c-sy}^{(L)}$ and $H_{c-se}^{(L)}$ to obtain the final representation.

$$H_c^{(L)} = \frac{\alpha H_{c-sy}^{(L)} + \beta H_{c-se}^{(L)}}{2} \quad (19)$$

Where α and β are trainable parameters, $H_c^{(L)} \in R^{n \times d_{gcn}}$.

3.5 Feature fusion

Before fusing the representation extracted by specific and common convolutional modules, we use the mask mechanism and average pooling to capture the aspect vectors h_{sy} , h_c and h_{se} from $H_{sy}^{(L)}$, $H_{se}^{(L)}$ and $H_c^{(L)}$. Then the final aspect representation is obtained by concatenating h_{sy} , h_c and h_{se} . The formulas are described as:

$$h_{sy} = f(mask(H_{sy}^{(L)})) \quad (20)$$

$$h_c = f(mask(H_c^{(L)})) \quad (21)$$

$$h_{se} = f(mask(H_{se}^{(L)})) \quad (22)$$

$$h_a = [h_{sy}; h_c; h_{se}] \quad (23)$$

where $mask(\cdot)$ represents the masked function, which means that only aspect vectors are activated in $H_{sy}^{(L)}$, $H_{se}^{(L)}$ and $H_c^{(L)}$, the context vectors are discarded. Mask function is formulated as follow:

$$\text{mask} = \begin{cases} 0, & 1 \leq t < \tau + 1, \tau + m < t < n \\ 1, & \tau + 1 \leq t \leq \tau + m \end{cases} \quad (24)$$

As described in section 3.1: $\tau + 1 \leq t \leq \tau + m$ denotes the aspect words index. $f(\cdot)$ is the average pooling function. The representation h_a is simply concatenated with h_{sy} , h_c and h_{se} , however, it could not distinguish the importance of syntax, semantics and their combination. Considering to reflect the significance of each part, a MLP layer is

utilized to learn the adaptive important weights of the representations and fuse them simultaneously. As described in section 3.1: $\tau + 1 \leq t \leq \tau + m$ denotes the aspect words index. $f(\cdot)$ is the average pooling function. The representation h_a is concatenated with h_{sy} , h_c and h_{se} , however, it could not distinguish the importance of syntax, semantics and their combination. Considering to reflect the significance of each part, a MLP layer is utilized to learn the adaptive weights of the representations and fuse them simultaneously.

$$\hat{h}_a = \text{ReLU}(h_a W_1^T + b_1) \quad (25)$$

By training the learnable weight matrix $W_1^T \in R^{3d_{gcn} \times d_{gcn}}$, our proposed model is able to fuse syntactic, semantic and their combination representations adaptively and derive deeper correlation information for sentiment classification.

Model Training

The final aspect representation \hat{h}_a is sent to a Soft-Max layer to calculate the probability distribution of the sentiment polarities as follows:

$$\tilde{y} = \text{softmax}((\hat{h}_a) W_2^T + b_2) \quad (26)$$

Considering that the embeddings $H_{sy}^{(L)}$ and $H_{c-sy}^{(L)}$ are both learned from graph (A_{sy}, H^c) , we hope that different information can be captured by them. Thus, the difference loss between $H_{sy}^{(L)}$ and $H_{c-sy}^{(L)}$ is defined as $\|H_{sy}^T H_{c-sy}\|_{l2}^2$. The smaller their inner product is (i.e., orthogonal constraint), the better result is obtained. Similarly, $\|H_{se}^T H_{c-se}\|_{l2}^2$ is the difference loss between $H_{se}^{(L)}$ and $H_{c-se}^{(L)}$. We set the difference loss \mathcal{L}_d as:

$$\mathcal{L}_d = \|H_{sy}^T H_{c-sy}\|_{l2}^2 + \|H_{se}^T H_{c-se}\|_{l2}^2 \quad (27)$$

In addition, $H_{c-sy}^{(L)}$ and $H_{c-se}^{(L)}$ are the outputs of *Common-GCN*, to confirm that common features are learnt, the smaller the Euclidean distance is, the better representation is obtained. We use \mathcal{L}_s constraint to enhance their similarities:

$$\mathcal{L}_s = \|H_{c-sy} - H_{c-se}\|_{l2}^2 \quad (28)$$

Finally, the cross entropy loss with L_2 regularization, as shown in equation(29), all parameters of our proposed model is optimized via backpropagation.

$$L = -\sum_i \sum_{j=1}^P y_i^j \log \hat{y}_i^j + \lambda \|\theta\|^2 + \gamma \mathcal{L}_d + \delta \mathcal{L}_s \quad (29)$$

Table 1: Statistics of datasets

| Dataset | Positive | | Negative | | Neutral | |
|---------|----------|------|----------|------|---------|------|
| | Train | Test | Train | Test | Trtrain | Test |
| Rest14 | 2164 | 728 | 807 | 196 | 637 | 196 |
| Lap14 | 994 | 341 | 870 | 128 | 464 | 169 |
| Twitter | 1561 | 173 | 1560 | 173 | 3127 | 346 |

where i is the index of i^{th} sample, j is the index of j^{th} sentiment class, P is the number of sentiment classes, y is the real distribution of sentiment and \hat{y} is the predicted one. Besides, λ is the weight of L_2 regularization, θ represents all parameters in the model. γ and δ are hyperparameters of the consistency and disparity constraint terms.

4 Experiment

In this section, we will first introduce the benchmark datasets for evaluation and the baseline models utilized for comparison. Secondly, the experimental implementation and parameter setting will be reported. Then the results of different experiments are analyzed from different perspectives including the comparison with baseline models, ablation study, the effect of hyper-parameters. Finally, some representative cases are discussed.

4.1 Datasets

Our model is evaluated the performance on SemEval 2014(Pontiki et al., 2014) and Twitter dataset provided by (Adaptive recursive neural network for target-dependent twitter sentiment classification), which are authoritative evaluation dataset in ABSA. The statistics of the datasets in Table 1.

4.2 Baseline Methods

We have chosen some mainstream and lasted models in ABSA for comparison, including:

Semantic-based models: ATAE-LSTM)(Wang et al., 2016b), IAN (Ma et al., 2017), RAM (Chen et al., 2017), MGAN (Fan et al., 2018).

Syntax-based models: LSTM+SynATT (He et al., 2018b), AdaRNN (Dong et al., 2014), PhraseRNN (Nguyen and Shirai, 2015), PWCON-Dep (Zhang et al., 2019b), AS-GCN (Zhang et al., 2019a), CDT (Sun et al., 2019), TD-GAT (Huang and Carley, 2019), R-GAT (Wang et al., 2020a), DGEDT (Tang et al., 2020) and Repwalk (Zheng et al., 2020)

Table 2: Sentiment classification results

| Category | Method | Twitter | | Lap14 | | Rest14 | |
|-------------|-----------------------------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | | Accuracy | Macro-F1 | Accuracy | Macro-F1 | Accuracy | Macro-F1 |
| Syn. | AdaRNN(Dong et al.,2014) | 66.30 | 65.90 | - | - | - | - |
| | PhraseRNN(Nguyen and Shirai,2015) | - | - | - | - | 66.20 | 59.32 |
| | LSTM+SynATT(He et al.,2018a) | - | - | 72.57 | 69.13 | 80.45 | 71.26 |
| | PWCN-Dep (Zhang et al.,2019) | - | - | 76.12 | 72.12 | 80.96 | 72.21 |
| | ASGCN (Zhang et al.,2019) | 72.15 | 70.40 | 75.55 | 71.05 | 80.77 | 72.02 |
| | CDT (Sun et al.,2019b) | 74.66 | 73.66 | 77.19 | 72.99 | 82.30 | 74.02 |
| | TD-GAT (Huang and Carley,2019) | - | - | 74.00 | - | 80.35 | - |
| | R-GAT (Wang et al.,2020) | 75.57 | 73.82 | 77.42 | 73.76 | 83.30 | 76.08 |
| | RepWalk (Zheng et al.,2020) | 74.4 | 72.6 | 78.2 | 74.3 | 83.8 | 76.9 |
| Att. | DGEDT (Tang et al.,2020) | 74.8 | 73.4 | 76.8 | 72.3 | 83.9 | 75.1 |
| | ATAE-LSTM (Wang et al.,2016b) | - | - | 68.70 | - | 77.20 | - |
| | IAN (Ma et al.,2017) | - | - | 72.10 | - | 78.60 | - |
| | RAM (Chen et al.,2017) | 69.36 | 67.30 | 74.49 | 71.35 | 80.23 | 70.80 |
| Ours | MGAN (Fan et al.,2018) | 72.54 | 70.81 | 75.39 | 72.47 | 81.25 | 71.94 |
| | DM-GCN | 76.93 | 75.9 | 78.48 | 74.9 | 83.98 | 75.59 |
| | R-GAT-BERT(Wang et al.,2020) | 76.15 | 74.88 | 78.21 | 74.07 | 86.60 | 81.35 |
| BERT | DGEDT-BERT(Tang et al.,2020) | 77.9 | 75.4 | 79.8 | 75.6 | 86.3 | 80.0 |
| | DM-GCN-BERT | 78.06 | 77.36 | 80.22 | 77.28 | 87.66 | 82.79 |

* The symbol ‘-’ indicates this result is not available in their work.

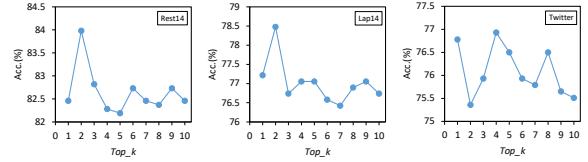
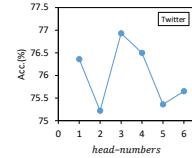
4.3 Implementation and parameter settings

For fairness and validation, the best results of compared models are used in models comparation. The Stanford parser (<https://stanfordnlp.github.io/CoreNLP/>) is utilized to get syntactic dependency tree in our work. Inspired by CDT (Sun et al., 2019), we exploit 300-dimensional Glove vectors (Pennington et al., 2014) for the word embeddings. Besides, on purpose to enrich the sentence representation, a 30-dimensional part-of-speech (POS) embeddings and 30-dimensional position embeddings are concatenated to word embeddings. The rest of parameter settings is given in the code¹.

4.4 Result and Analysis

4.4.1 Overall Performance Comparison

The results of comparison with all baseline models are shown in Table2. First, compared with the lasted syntactic-based model DGEDT, Repwalk, and R-GAT, DM-GCN significantly improves the accuracy in sentiment classification, which indicates that DM-GCN performs better at encoding the syntactic and semantic information by fusing them adaptively. Second, as the first proposed concept, the common information shared by semantic and syntactic spaces, has been proved its effective in ABSA, which can be judged from the improvement of accuracy on three datasets. The results have proved that only encode semantic or syntac-

Figure 4(a): Impact of k Figure 4(b): Impact of $head - numbers$

tic information is inadequate in aspect-based sentiment classification. On the contrast, fusing the semantic and syntactic information and their combination optimally, just as humanity brain does, is the key to the better performance of our DM-GCN.

4.4.2 Ablation Study

In this section, we further conduct an ablation study to verify the validity of each module in our DM-GCN. The result is in Table 4. The basic DM-GCN is regarded as a baseline model..

First, removal of syntactic information (i.e.DM-GCN w/o H_{sy}) leads to the drop of accuracy drop of 1.42%, 1.11% and 1.65% on Twitter, Lap14 and Rest14 respectively, which demonstrates the significance of syntactic information and verifies syntactic information plays different roles in dif-

¹<https://github.com/pangsg/DM-GCN>

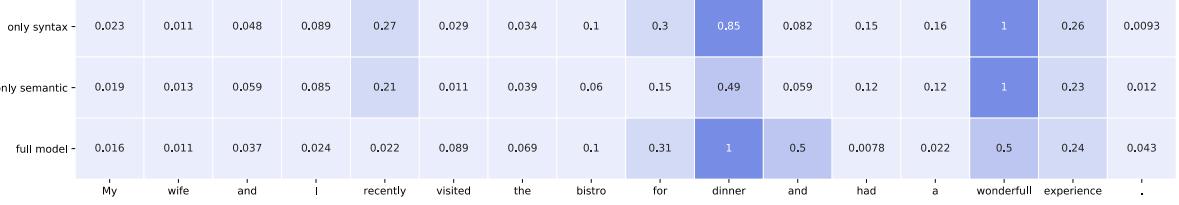


Figure 5: The word relevance scores. The aspect word is ‘dinner’. The ground truth is ‘neutral’ while the prediction of the only syntax, only semantic and full model are ‘positive’, ‘positive’ and ‘neutral’ respectively.

ferent sentence of three datasets. Second, after we get rid of the semantic information (i.e. DM-GCN w/o H_{se}), the model ranges the lowest accuracy on three datasets, which indicates that the dynamic multi-head self-attention mechanism proposed by us is able to capture the most correlated semantic information between aspect and opinion words. As for ‘DM-GCN w/o H_{common} ’ model, the accuracy decreases 1.3%, 2.54% and 1.63% on Twitter, Lap14 and R Rest14 respectively. Recall the main reason on Lap14, we conclude that the common information mixes up of the characteristics of syntactic and semantic information, which is significant to Lap14 datasets. As a result, the ablation experimental outcomes confirm the contribution of both components.

Table 3: Ablation study results

| Model | Twitter Accuracy | Lap14 Accuracy | Rest14 Accuracy |
|-------------------------|------------------|----------------|-----------------|
| DM-GCN | 76.93 | 78.48 | 83.93 |
| DM-GCN w/o H_{se} | 74.94 | 75.79 | 81.92 |
| DM-GCN w/o H_{sy} | 75.51 | 77.37 | 82.28 |
| DM-GCN w/o H_{common} | 75.63 | 75.94 | 82.3 |

* ‘DM-GCN w/o H_{sy} ’: removing syntactic convolution module;
 ** ‘DM-GCN w/o H_{se} ’: removing semantic convolution module;
 *** ‘DM-GCN w/o H_{common} ’: removing common convolution module.

4.4.3 Hype-parameters Analysis

The results of the research about the hype-parameter k and the head numbers of our proposed semantic convolutional module are in Figure 4.

1. Effect of hype-parameter k . From Figure 4a, there are some observation can be noted:1) The best performance appears when k is equals to 3,3 and 4 respectively in Rest14, Lap14 and Twitter datasets. The accuracy decreases on three datasets when k becomes bigger. We conclude that the bigger k we set, the more noise is aggerated from irrelevant words in the sentence, which interferes with our model’s ability to determine the sentiment polarity of sentences.

2. Effect of hype-parameter $head\text{-}number$. Take

Twitter dataset for example. The result in Figure 4b show the impact of head-numbers in multi-head self-attention mechanism of our models. Intuitively, the best performance appears when head number is equal to 3. Considering the characteristics of multi-head self-attention, we take the attitude that the model becomes haphazard and overfitting when head number decreases and increases respectively.

4.4.4 Case Study and Attention Distribution

In this section, we utilize the method “Mask Experiment” proposed by CDT(Sun et al., 2019)to estimate the sentiment contribution of word w in the sentence a .By leveraging this method,we evaluate the effective of our proposed semantic and syntactic convolution module mathematically. The formula is as follows:

$$\gamma(w, s) = \frac{1}{m} \sum_{i=1}^d |\hat{h}_a^i - \hat{h}_{(a/w)}^i| \quad (30)$$

As described in CDT(Sun et al., 2019),where d is the dimension of the final representation \hat{h}_a learned by our model, w is the masked word, which indicates w becomes zero vector, $\hat{h}_{(a/w)}$ is the final representation of sentence a generated by our DM-GCN with the word w masked. If $\hat{h}_a^i = \hat{h}_{(a/w)}^i$, which demonstrates the word w has no sentiment contribution to sentence a .

Figure 5 is the visualization of the attention placed on words, from which can be observed that when syntactic or semantic convolution module (1st or 2nd row) is utilized separately, the model has assigned the highest attention on “wonderful” mistakenly. Regarding “wonderful” as the opinion word of aspect “dinner”, which results in the wrong judgment of the sentence eventually. However, our proposed DM-GCN (3rd row) is able to adaptively reduce the attention on irrelevant word “wonderful” and increase the score on “dinner” adaptively. Implying that our DM-GCN captures the most correlated information from syntactic, semantic and

their combination spaces, which verifies the validity of our strategy mentioned before.

5 Conclusion

In this paper, we have proposed a dynamic and multi-channel graph convolution network to encoding the syntactic and semantic information for aspect-based sentiment analysis. Motivated by the human brain’s mechanisms of language, we study how to adaptively learned the most correlated information from syntactic, semantic, and their combination spaces and fuse them sufficiently according to the characteristic of sentences. The experimental results on three benchmark datasets showed that we have achieved superior performance over the baseline methods. Besides, the ablation study and case analysis have exploited the influence of hyperparameters and validated the role of each component in our proposed model.

Acknowledgments

This work was supported by the National Statistical Science Research Project of China under Grant No. 2016LY98, the Science and Technology Department of Guangdong Province in China under Grant Nos. 2016A010101020, 2016A010101021 and 2016A010101022, the Characteristic Innovation Project of Guangdong Colleges and Universities (No. 2018KTSCX049), the Science and Technology Plan Project of Guangzhou under Grant Nos. 201802010033 and 201903010013.

References

- Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.
- 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.
- Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. 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)*, pages 49–54.
- Feifan Fan, Yansong Feng, and Dongyan Zhao. 2018. Multi-grained attention network for aspect-level sentiment classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3433–3442.
- Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018a. Effective attention modeling for aspect-level sentiment classification. In *Proceedings of the 27th international conference on computational linguistics*, pages 1121–1131.
- Shexia He, Zuchao Li, Hai Zhao, and Hongxiao Bai. 2018b. Syntax for semantic role labeling, to be, or not to be. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2061–2071, Melbourne, Australia. Association for Computational Linguistics.
- Binxuan Huang and Kathleen M Carley. 2019. Syntax-aware aspect level sentiment classification with graph attention 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 5472–5480.
- Thomas N Kipf and Max Welling. 2019. Semi-supervised classification with graph convolutional networks.
- Cheng Li, Xiaoxiao Guo, and Qiaozhu Mei. 2017. Deep memory networks for attitude identification. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 671–680.
- Liina and Pylkknen. 2019. The neural basis of combinatorial syntax and semantics. *Science (New York, N.Y.)*, 366(6461):62–66.
- Kang Liu, Heng Li Xu, Yang Liu, and Jun Zhao. 2013. Opinion target extraction using partially-supervised word alignment model. In *IJCAI*, volume 13, pages 2134–2140.
- Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4068–4074.
- Thien Hai Nguyen and Kyoaki 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.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.
- Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the*

- 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 27–35, Dublin, Ireland. 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 5683–5692.
- 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.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.
- Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2018. Graph attention networks. *stat*, 1050:4.
- Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020a. Relational graph attention network for aspect-based sentiment analysis. *arXiv preprint arXiv:2004.12362*.
- 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.
- Xiao Wang, Meiqi Zhu, Deyu Bo, Peng Cui, Chuan Shi, and Jian Pei. 2020b. Am-gcn: Adaptive multi-channel graph convolutional networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1243–1253.
- 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.
- Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. 2019. Bert post-training for review reading comprehension and aspect-based sentiment analysis. *arXiv preprint arXiv:1904.02232*.
- Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.
- Chen Zhang, Qiuchi Li, and Dawei Song. 2019a. 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 4560–4570.
- Chen Zhang, Qiuchi Li, and Dawei Song. 2019b. Syntax-aware aspect-level sentiment classification with proximity-weighted convolution network. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1145–1148.
- Yaowei Zheng, Richong Zhang, Samuel Mensah, and Yongyi Mao. 2020. Replicate, walk, and stop on syntax: An effective neural network model for aspect-level sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9685–9692.