Structure Graph Refined Information Propagate Network for Aspect-Based Sentiment Analysis

Weihao Huang, School of Electronics and Information Engineering, South China Normal University, China
Shaohua Cai, Center for Faculty Development, South China Normal University, China*
Haoran Li, School of Electronics and Information Engineering, South China Normal University, China
Qianhua Cai, School of Electronics and Information Engineering, South China Normal University, China*

ABSTRACT

The main task of aspect-based sentiment analysis is to determine the sentiment polarity of a given aspect in the sentence. A major issue lies in identifying the aspect sentiment is to establish the relationship between the aspect and its opinion words. The application of syntactic dependency trees is one such resolution. However, the widely-used dependency parsers still have challenges in obtaining a solid sentiment classification result. In this work, an information propagation graph convolutional network based on syntactic structure optimization is proposed on the task of ABSA. To further complement the syntactic information, the semantic information is incorporated to learn the representations using graph information propagation mechanism. In addition, the effects of syntactic and semantic information are adapted via feature separation. Experimental results on three benchmark datasets show that the proposed model achieves satisfying performance against the state-of-the-art methods, indicating that the model can precisely build the relation between aspect and its context words.

KEYWORDS

Aspect-Based Sentiment Analysis, Bert, Graph Convolutional Networks, Information Propagate, NLP, Semantic Graph, Structure Graph, Structure Induction

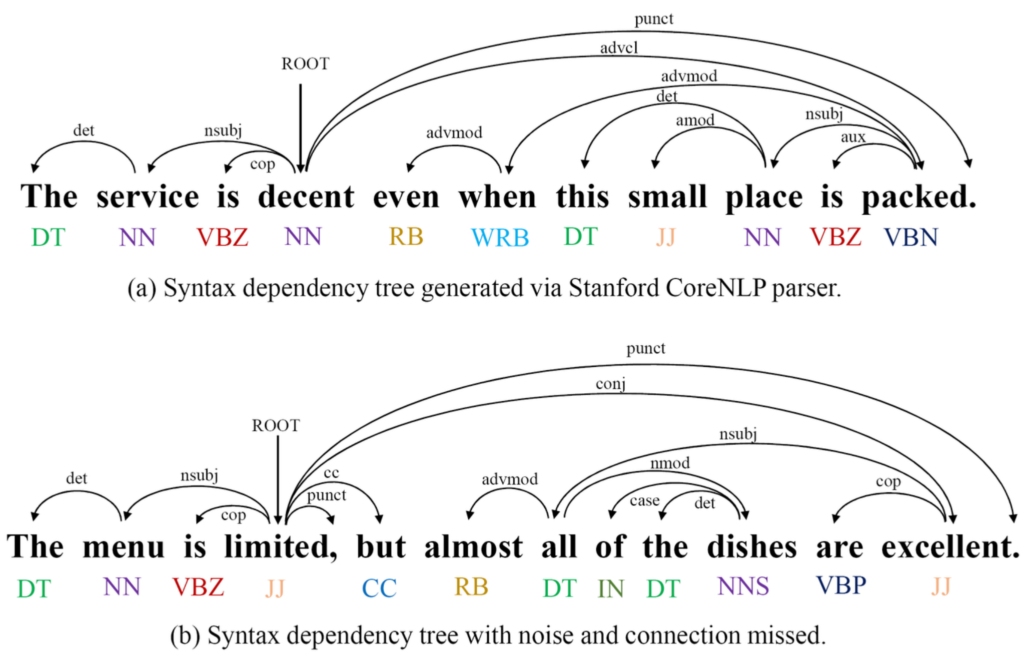
INTRODUCTION

Data mining sets a foundation of natural language processing. In our daily lives, people are constantly invited to share their opinions and preferences with the rest of the world, which results in an explosion of textual information. As such, data mining provides an opportunity to deal with the opinions on products, stocks, policies, and everything. In this context, sentiment analysis is thereby developed to determine the opinion of people regarding a given topic via textual data mining. Aspect-based sentiment analysis (ABSA) is currently an ongoing trend for precisely mining the user's opinion.

DOI: 10.4018/IJDWM.321107
*Corresponding Author

Comprehensively, ABSA is a fine-grained sentiment analysis task in the field of sentiment analysis (Tang et al., 2014; Yang et al., 2017). The main purpose of ABSA is to identify the sentiment polarity (i.e., positive, neutral, or negative) toward a given aspect in a sentence or document. For instance, in the sentence 'The service is decent even when this small place is packed,' (Figure 1a), the two aspects 'service' and 'place' are extracted, whose sentiment polarities are classified as negative and positive, respectively. Aiming to deal with the issue of multiple aspects within one sentence, the interaction between an aspect and its contexts has to be resolved. The description of speech and dependency relationship is shown in Table 1.

Figure 1.
Two examples of syntax dependency tree



(a) Syntax dependency tree generated via Stanford CoreNLP parser.

(b) Syntax dependency tree with noise and connection missed.

Table 1.
Description of parts of speech and dependencies

Tag	Description	Dependencies	Description
DT	Determiner	ROOT	The most important word
NN	Noun, singular or mass	det	determiner
VBZ	Verb, 3rd person singular present	nsubj	nominal subject
RB	Adverb	cop	copula
WRB	wh-adverb	cc	coordination
JJ	Adjective	advmod	adverb modifier
VBN	Verb, past participle	conj	conjunct
CC	Coordinating conjunction	punct	punctuation
IN	Preposition or subordinating conjunction	nmod	noun modifier
NNS	Noun, plural		
VBP	Verb, non-3rd person singular present		

Theoretically, a sentence has distinguishing representations in both syntax space and semantic space. In order to clarify the distinctiveness, the syntax and the semantics of the sentence are taken as supplementary to each other. With the fusion of both categories of information, the sentiment information can be precisely delivered for further processing. Encouragingly, the wide application of graph convolutional networks (GCNs) reports profound effects and gives rise to new opportunities in ABSA domain (Kipf & Welling, 2016). More recently, GCNs show the superiority in extracting syntactic information, which is further integrated with the aspect representation for sentiment analysis (Zhang et al., 2018; Sun et al., 2019; Huang & Carley, 2019; Tang et al., 2020; Zhang & Qian, 2020). Notwithstanding, the noise that comes from the irrelevant contextual information can still be introduced during the modelling of basic syntax dependency trees. Thereby, the incorporating of unrelated information has an impact on the sentiment classification results.

For the purpose of improving the working performance of an ABSA, two main focuses are highlighted. On the one hand, the structure of the original syntax dependency tree has to be modified, based on which the sentence dependencies can be parsed in a more comprehensive way. In such a manner, the noise can be effectively removed while more related information can be extracted. Recent publications provide their modification on syntax dependency trees on the task of ABSA. Zhang and Qian (2020) establish multiple subgraphs to characterize different types of dependencies within the syntax dependency tree. Wang et al. (2020) propose a relational graph attention network for the pruning of the basic syntax dependency tree, in which way an aspect-oriented dependency tree is constructed. Chen et al. (2020) generate the aspect representations by fusing the dependency graphs and the target-specific latent graphs.

The main focus of current approaches lies in addressing the dependency within large amounts of information and thus improve the sentiment classification accuracy. Whereas these approaches still have challenges in distilling the connecting relation among words during syntactic parsing, according to the parsing of the syntax dependency tree in Figure 1b, the dependency between ‘dishes’ and ‘excellent’ is neglected, while the syntactical connection between ‘limited’ and ‘but’ makes no sense. By contrast, the refinement on the basic syntax dependency tree benefits the capturing of both local and global interaction among words in the sentence. On the other hand, the integration of the syntax and the semantics is most pronounced, which aims to sufficiently exploit the key information.

The objective of this work is to reveal the exact relation between aspect and its contexts and address the interaction between syntax and semantics. A Latent Structure Refinement (LSR) method on the foundation of an iteratively refining strategy is proposed. In our model, the aspect-specific dependencies are established to demonstrate the relationship among words. Further, a feature propagation module is devised to deal with the communication of syntactic graph and semantic graph. In addition, an average pooling and feature separation module is built to obtain the representations of both the aspect and the contexts. Notably, two hyperparameters are introduced to denote the distinctiveness of the syntax and the semantics. The working performance of the proposed model is investigated on publicly available data sets. To sum up, the contributions of this work are threefold and can be summarized as follows:

1. Aiming to eliminate the noise generated during syntactic information parsing, a method for reshaping the basic syntax dependency tree is developed by using the iteratively refining strategy. With the pruning of dependencies, the relation between aspect and its context words is formed, and the syntactic information can thus be captured.
2. For the joint learning of the syntax and the semantics, the feature propagating module is worked out. In this way, the interaction between the syntactic information and the semantic information is enhanced and employed to obtain the aspect representation.
3. On the task of ABSA, experiments are conducted to evaluate the working performance of our model. Experimental results reveal that LSR is a competitive alternative compared with the state-of-the-art methods.

RELATED WORK

These years, we have witnessed the progression of applying syntax and semantics to describing the relation between the aspect and its context in deep learning-based ABSA approaches. The integration of syntactic information and semantic information becomes a primary choice in settling ABSA tasks and continues to achieve satisfying results.

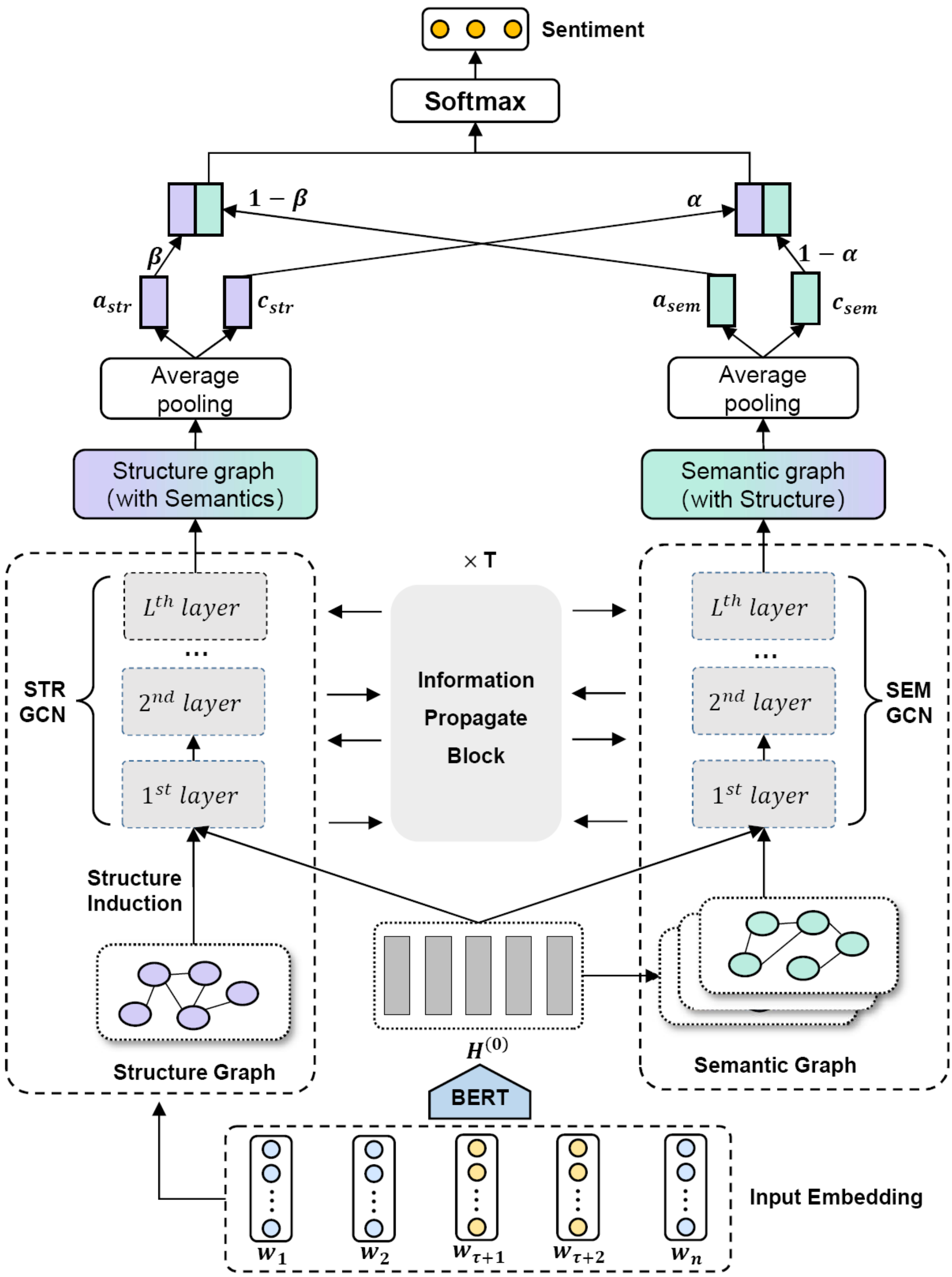
For one thing, the attention-based deep neural networks show the superiority in tackling semantic information by assigning distinguishing attentive weights to different context words. Previous work focused on integrating attention mechanisms into recurrent neural networks (RNNs) or convolution neural networks (CNNs) (Yang et al., 2017; Liu & Zhang, 2017; Zeng et al., 2019). Chen et al. (2017) designed the multi-head attention network that aims to capture the long-range sentiment features within contexts. Xue and Li (2018) integrated the attention mechanism and the gating unit into CNNs to extract aspect-based semantic information from the contexts. Notably, the pretrained model Bidirectional Encoder Representations from Transformer (BERT) (Devlin et al., 2018) also shows an impressive performance in dealing with ABSA tasks. Sun et al. (2019) constructed an aspect-based auxiliary sentence to transform ABSA tasks into sentence-pair classification tasks. Xu et al. (2019) proposed a post-training BERT method to optimize the parameter fine-tuning in ABSA tasks. For another, the sentence syntax is also leveraged in maintaining the dependencies among words and removing the distance between the aspect and its opinion word. Zhang et al. (2019b) proposed a weighted convolutional neural network to establish a syntax-aware context representation of the aspect, in order to clarify the syntactic weights between the aspect and its context words.

The employment of GCNs has brought about paradigm shifts to ABSA tasks. Research is still ongoing to fuse the syntax and semantic into GCN-based approaches (Sun et al., 2019; Huang & Carley, 2019; Zhang et al., 2019a). A widely used manner is to transform the syntax dependency tree into a syntactic graph, based on which the sentiment information can be propagated to the aspect. The GAT(Graph Attention Networks) is developed for the reconstruction of a syntax dependency tree (Wang et al., 2020). To distinguish different types of relationships, the type-aware graph convolutional networks (T-GCNs) are proposed to explicitly utilize dependency types for ABSAs (Tian et al., 2021). Tang et al. (2020) devised a dependency graph-enhanced dual-transformer network (DGEDT) by jointly considering the flat representations learnt from Transformer and graph-based representations learnt from the corresponding dependency graph in an iterative interaction way (Tang et al., 2020). Zhang and Qian (2020) conducted convolution on hierarchical syntactic and lexical graphs to discriminate various types of dependency relations or lexical word pairs. Zhou et al. (2020) introduced a knowledge graph and a syntactic graph for joint modelling, based on which the sentiment classification is performed via a multi-head attention mechanism. In the context of syntax-based GCNs and semantic-based GCNs, Li et al. (2021) built a differential regularizer to learn both categories of information sufficiently.

PROPOSED MODEL

Figure 2 presents the architecture of the proposed model. A given sentence is first sent to the encoder to derive the sentence embedding and obtain the sentence hidden states using BERT. The hidden layer representation is used for semantic encoding and thus to generate the semantic graph. Besides, the sentence syntax is characterized via a syntax dependency tree, which is pruned for further processing. The GCNs are employed to encode both the syntax and the semantics. Moreover, the information of each GCN is propagated between each other to enhance the learning of syntactic and semantic features. Average pooling and feature separation are performed on the outcomes of GCNs in sequence, based on which the representations of the aspect and the context can be established. The final representation for sentiment analysis is the concatenation of the aspect representation and the context representation. More details of each working module in our model are described in detail as follows.

Figure 2.
Overview of SGRPN Model (SGRPN stands for structure graph refined information propagate network, STRGCN stands for structure graph convolutional network, SEMGCN stands for semantic graph convolutional network)



Preliminaries

The matrix-tree theorem is introduced in advance. A structured attention network is known as an effective approach for embedding categorical inference within a deep neural network (Kim et al., 2017). The matrix-tree theorem (Koo et al., 2007) is one such mechanism, which is first proposed for the working performance optimization of structured prediction models. Liu et al. (2019) applied the variant of matrix tree to single-document summarization in the field of NLP. In line with its working principle, the matrix-tree theorem can be used for syntax inference and refinement in ABSA tasks. Motivated by Nan et al. (2020), we take a syntactic graph as a relation variable, where the relationship among words can be derived in an end-to-end manner. In this way, a variant of the Kirchhoff's matrix tree theorem (Tutte, 2001) is utilized to determine the dependencies within the sentence. According to Tutte (2001), the basic matrix-tree theorem is proposed to resolve the issue of undirected spanning trees in undirected graphs. With respect to the structured predicting methods, computing the summation of weights and directed spanning trees is necessary. We thus defined a graph as $G = \left(V, E, A\right)$, where $V = \left\{0, 1, 2, \cdots, n\right\}$ represents the set of all nodes, E refers to the set of edges between nodes, and A is the set of edge weights. The Laplacian matrix (Tutte, 2001) of graph G can be written as:

$$L_{h,m}\left(\theta\right) = \begin{cases} \sum_{h'=1}^{n} A_{h',m}\left(\theta\right) & if\, h = m \\ -A_{h,m}\left(\theta\right) & otherwise \end{cases} \tag{1}$$

where $h, m = 1 \ldots n$.

For a matrix X, let $X^{\left(h,m\right)}$ be the vector of the h-th row and the m-th column. A determinant of X can be generated by removing the h-th row and the m-th column from X. Then, the weight of any directed spanning tree within G is the product of edge weights $A_{h,m}\left(\theta\right)$ (Tutte, 2001, p. 140). Notably, only if crossing the columns of the Laplace operator (Tutte, 2001) can the signs of the matrix complement vary slightly (Tutte,2001, p.150), which is:

$$\forall h, m (-1)^{h+m} L^{\left(h,m\right)}\left(\theta\right) = L^{\left(m,m\right)}\left(\theta\right) \tag{2}$$

Sentence Encoder

To start with, the BERT encoder is employed for sentence encoding and feature extraction. Given an n-word sentence $s = \left[w_1, w_2, \cdots, w_{\tau+1}, \cdots, w_{\tau+m}, \cdots, w_n\right]$, we take w_k to denote the word embedding of the k-th word, where $k \in \left[1, n\right]$. The m-word aspect within is written as $\left[w_{\tau+1}, \cdots, w_{\tau+m}\right]$. The CLS vector is extracted as the sentence representation vector, based on which the sequence '[CLS] s [SEP] a [SEP]' is generated to obtain the representation vector of the aspect (Song et al., 2019). The word-level representation is subdivided, conforming to the first token embedding of each word in the sentence. Thereby, the sentence hidden state is derived as $H = \left\{h_1, h_2, \cdots, h_{\tau+1}, \cdots, h_{\tau+m}, \cdots, h_n\right\}$. The hidden layer representation of the aspect, that is, $\left[h_{\tau+1}, \cdots, h_{\tau+m}\right]$ is applied to the semantic graph generation.

Latent Structure Refinement

The sentence hidden state can be used to construct a syntactic graph (Sun et al., 2019). Each node represents a word in the sentence, while each edge indicates the dependency between two words. The adjacency matrix is denoted as:

$$A_{str}\left(i, j\right) = \begin{cases} 1 & if\, i \leftrightarrow j \\ 0 & others \end{cases} \tag{3}$$

where ‘ $i \leftrightarrow j$ ’refers to the dependency of node ‘ i ’ and node ‘ j.’

One can easily see that the basic syntactic graph contains noise from irrelevant words. For this reason, a structure induction module is developed, targeting at structure inference and refinement using an iteratively refining strategy (see Figure 3).

Let w_i and w_j be the node representations of nodes i and j. The nonnormalized attentive score between w_i and w_j can be computed via two feedforward neural networks and a bilinear transformation, which is:

$$f_{ij} = \left(\tanh\left(W_p w_i\right)\right)^T W_b \left(\tanh\left(W_c w_j\right)\right) \tag{4}$$

where $W_p \in \mathbb{R}^{d\times d}$ and $W_c \in \mathbb{R}^{d\times d}$ separately stand for the weighted coefficients of the two feedforward neural networks, d is the node representation dimension, $tanh$ represents the activation function, and $W_b \in \mathbb{R}^{d\times d}$ is the weight of the bilinear transformation.

Then, the probabilities for each edge within the syntactic structure can be calculated (Koo et al., 2007; Nan et al., 2020). For the graph G containing n nodes, a nonnegative weight is assigned to every single edge in the first place. We have:

$$\tilde{A}_{ij} = \begin{cases} 0 & if \;\; i = j \\ \exp\left(f_{ij}\right) & otherwise \end{cases} \tag{5}$$

where A_{ij} stands for the weight between w_i and w_j.

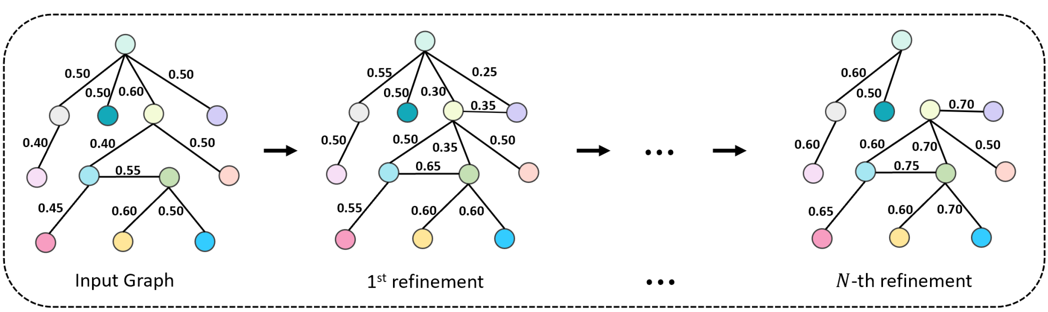
Likewise, the root fraction of node w_i can be derived, which indicates the nonnormalized probability of root node w_i:

$$f_i^r = W_r w_i \tag{6}$$

where $W_r \in \mathbb{R}^{d\times d}$ is the weight of the linear transformation.

Subsequently, the Laplace matrix (Tutte, 2001) $L \in \mathbb{R}^{n\times n}$ of G is obtained, which is written as:

Figure 3.
Schematic of syntactic structure induction



$$L_{ij} = \begin{cases} \sum_{i'=1}^{n} \tilde{A}_{i'j} & if \;\; i = j \\ -\tilde{A}_{ij} & otherwise \end{cases} \tag{7}$$

together with:

$$\widehat{L}_{ij} = \begin{cases} \exp\left(f_i^r\right) & if \;\; i = 1 \\ L_{ij} & if \;\; i > 1 \end{cases} \tag{8}$$

where $\widehat{L} \in \mathbb{R}^{n \times n}$ is the variant of the basic Laplace matrix to facilitate processing.

In order to derive the edge probability A_{ij} between w_i and w_j, we take the Kronecker delta (Tutte, 2001) as the δ operator. Therefore, A_{ij} is given by:

$$\begin{aligned} A_{ij} \;\; &= \left(1 - \delta_{1,j}\right) \tilde{A}_{ij} \left[\widehat{L}^{-1}\right]_{ij} \\ &-\left(1 - \delta_{i,1}\right) \tilde{A}_{ij} \left[\widehat{L}^{-1}\right]_{ji} \end{aligned} \tag{9}$$

where $A \in \mathbb{R}^{n \times n}$ is the weighted adjacency matrix of the syntactic structure via derivation and optimization.

At this stage, the matrix is sent to the GCN for encoding and continuously updating the node representations. Notably, the GCN is employed to extract the syntactic information of the sentence, that is:

$$H_{str}^{(l+1)} = \sigma\left(A_{str} H_{str}^{(l)} W_{str}^{(l+1)}\right) \tag{10}$$

with $A_{str} = A, H_{str}^{(l+1)} \in \mathbb{R}^{k \times d}$ and the initialized input $H_{str}^{(0)} = H^{(0)}$. The matrix $W_{str}^{(l+1)} \in \mathbb{R}^{k \times d}$ is the weighted matrix of the l-th GCN layer, with d denoting the output dimension of the GCN.

Semantic Graph Generation

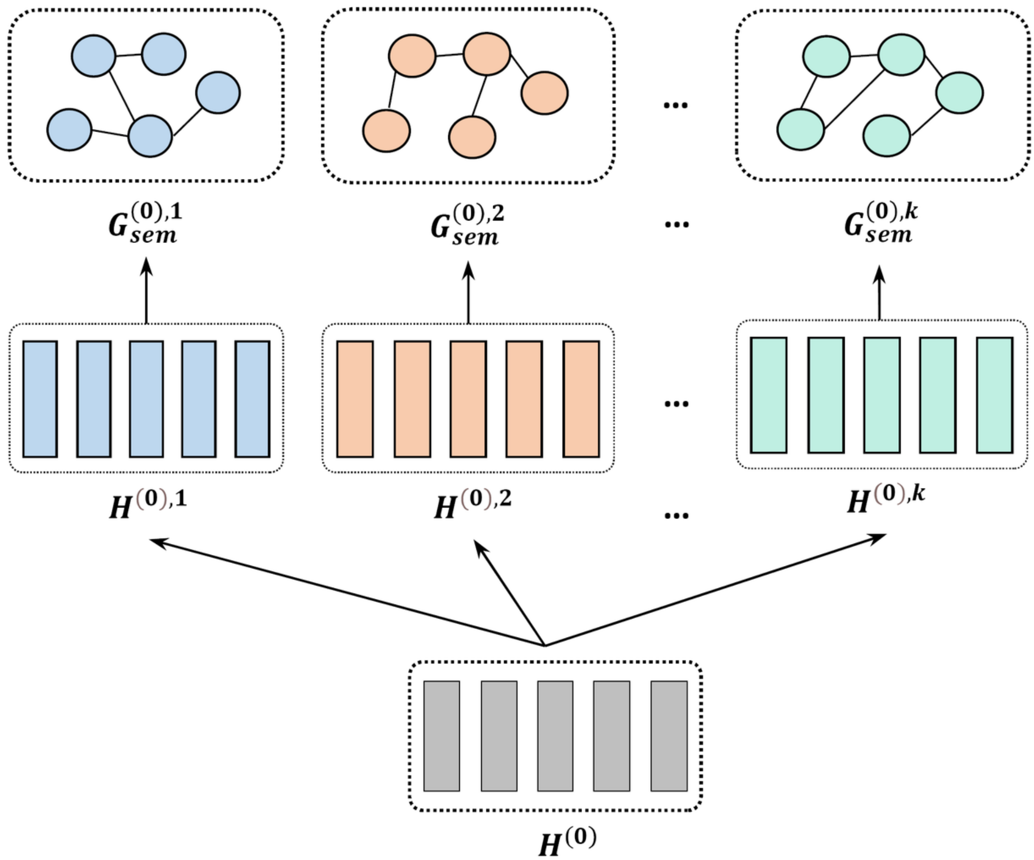
In addition to the optimization of syntactic structure, the processing on sentence semantics is carried out as well. In our model, a *k*-Head cosine similarity computation algorithm (Yan et al., 2021) is employed as shown in Figure 4, which aims to capture the semantic information. Specifically, the sentence hidden representation H derived from the BERT encoder is mapped into *K* semantic spaces of *d*-dimension, $G_{sem}^{(0),k}$ denotes the semantic graphs of different Spaces. We thus have:

$$H^{(0),k} = \sigma\left(H^{(0)} W_H^k + b_H^k\right) \tag{11}$$

with:

$$H^{(0),k} = \left[h_1^{(0),k}, h_2^{(0),k}, \cdots, h_n^{(0),k}\right], k \in \left[1, K\right] \tag{12}$$

Figure 4.
Overview of semantic graph generation



where $W_H^k \in \mathbb{R}^{d\times d}$ and $b_H^k \in \mathbb{R}^{1\times d}$ represent the mapping matrix and bias vector, respectively.

Subsequently, the semantic relation among words is derived using distance metric learning, based on which the semantic graph G_{sem} is obtained. The relation between node w_i and node w_j can be defined as:

$$A_{sem}\left[i,j\right] = \varphi\left(i,j\right) \tag{13}$$

together with:

$$\varphi\left(i,j\right) = \frac{1}{K}\sum_{k=1}^{K} a_{i,j}^{k} \tag{14}$$

$$a_{i,j}^{k} = \begin{cases} 1 \;\; if\, cos\left(h_i^{(0),k}, h_j^{(0),k}\right) > \rho \\ 0 \;\; otherwise \end{cases} \tag{15}$$

and $A_{sem} \in \mathbb{R}^{n \times n}$.

The semantic graph G_{sem} is fed into the GCN encoder to extract the semantic information of the sentence:

$$H_{sem}^{(l+1)} = \sigma\left(\tilde{A}_{sem} H_{sem}^{(l)} W_{sem}^{(l+1)}\right) \tag{16}$$

with $H_{sem}^{(l+1)} \in \mathbb{R}^{n \times d}$, and:

$$H_{sem}^{(0)} = H^{(0)} \tag{17}$$

where $W_{sem}^{(l+1)} \in \mathbb{R}^{n \times d}$ is the weighted matrix of the l-th layer in GCN.

Graph Information Propagation

The graph information propagating module is designed to learn the interaction between syntactic information and the semantic information. As mentioned in the work of Zhao et al. (2021), feature propagation aims to reveal the interaction in heterogeneous graphs. In such a manner, both the syntactic graph and the semantic graph can be updated. Concretely, the idea of information propagating is to transmit the similarity features to the spatially close neighbouring nodes. For example, if node A is in relation to node B while node B is in relation to node C, there can be a connection between node A and node C. As a result, the node information from both the syntactic graph and semantic graph can be fully interacted. That is, a relation is set to the nodes with similarities by mutual propagation between graphs.

We define the input of information propagation modules as $H_{str}^{(l+1)}$ and $H_{sem}^{(l+1)}$. With the utilization of syntax and semantics, the updated syntactic graph representation is generated as:

$$H_{str}^{(l+1)} = A_{str} H_{str}^{(l+1)} \tag{18}$$

$$H_{str} = H_{str}^{(l+1)} H_{sem}^{(l+1)} \tag{19}$$

Likewise, the updated syntactic graph representation can be:

$$H_{sem} = H_{sem}^{(l+1)} H_{str}^{(l+1)} \tag{20}$$

Since the syntactic graph and the semantic graph are continuously updated in each GCN layer, each propagation outcome is sent to the GCN for the iterative updating of information. The computation in Equations (19) and (20) is carried out L times repeatedly with $l \in \left[1, L\right]$. Hence, two categories of information propagating graphs, that is, H_{str} and H_{sem}, are obtained.

Average Pooling and Feature Separation

The representations of both the aspect and the context are established based on the syntactic information and the semantic information. The average pooling operation is conducted on H_{str} and H_{sem}, respectively. To improve the sentiment classification accuracy, we separate the syntactic features and the semantic features from vectors of aspect and the context. Taking the syntactic representation as an instance, the separation of syntactic features is performed by using the following formula:

$$a_{str} = AvgPool\left(\left[h_{str,i}\right]\ i \in \left(\tau, \tau + m\right]\right) \tag{21}$$

$$C'_{str} = \left[h_{str,j}\right]\ j \in \left[1, \tau\right] \cup \left(\tau + m, n\right] \tag{22}$$

$$Q'_{str} = softmax\left(a_{str} W_{str} C_{str}^{\ T}\right) \tag{23}$$

$$c_{str} = AvgPool\left(Q'_{str} \cdot C'_{str}\right) \tag{24}$$

where $AvgPool\left(\cdot\right)$ refers to the average pooling operation, $W_{str} \in \mathbb{R}^{d\times d}$ is the learnable weighted matrix, $Q'_{str} \in \mathbb{R}^{1\times d}$ is the weighting vector, $C_{str} \in \mathbb{R}^{(n-m)\times d}$, $a_{str} \in \mathbb{R}^{1\times d}$, and $c_{str} \in \mathbb{R}^{1\times d}$. In this way, the syntax-based aspect representation a_{str} and the context representation c_{str} can be obtained. Similarly, for the semantic representations $H_{sem}^{(1)}, H_{sem}^{(2)}, \ldots, H_{sem}^{(L)}$, we also obtain the semantic-based aspect representation a_{sem} and context representation c_{sem}.

The final representations are denoted as the concatenation of syntactic information and semantic information. We thus introduce two learnable hyperparameters, that is, α and β, together with a fully connected layer. Specifically, the parameter α and $\left(1-\alpha\right)$ is used for separately weighting the aspect and the contexts within the syntactic information. Likewise, the weighted representation of aspect and context based on semantic information can also be derived by using the parameter β:

$$h'_{context} = \alpha c_{str} + \left(1-\alpha\right) c_{sem} \tag{25}$$

$$h'_{aspect} = \beta a_{str} + \left(1-\beta\right) a_{sem} \tag{26}$$

The final representation for sentiment classification is the concatenation of the aspect representation and the context representation, which is:

$$h' = h'_{aspect} + h'_{context} \tag{27}$$

$$h = \sigma\left(h' W_s + b_s\right) \tag{28}$$

where $h \in \mathbb{R}^{1\times d}$, $h' \in \mathbb{R}^{1\times 2d}$, $\text{and} W_s$ are the trainable weighted matrix and b_s is the bias vector.

Sentiment Classification

The final representation h is sent to a fully connected layer for sentiment classification. We map h to an aspect space with C categories of sentiment:

$$z = hW_z^T + b_z \tag{29}$$

where W_z^T and b_z stand for the trainable weighting matrix and the bias, respectively. The probability of the i-th aspect of the j-th $\left(j\epsilon\left[1, C\right]\right)$ sentiment polarity is thus computed:

$$y_{ij}^{'} = \frac{exp\left(z_{ij}\right)}{\sum_{k=1}^{C} exp\left(z_{ik}\right)} \tag{30}$$

Model Training

The training of our model was performed using the standard gradient descent algorithm with cross entropy and L_2 regularization. The loss function is defined as:

$$L = -\sum_{i \in D} \sum_{j \in P} y_j^i \log \hat{y}_j^i + \lambda \left\| \theta \right\|_2 \tag{31}$$

where D is the training set, P is the class of sentiment polarity, y_j^i and $\hat{y}_j^i$ stand for the ground-truth sentiment distribution and the predicted one, respectively, λ is the regularization coefficient, and θ is the set of all trainable parameters of the proposed model.

EXPERIMENT

Data Sets

Experiments were carried out on three public data sets, that is, Twitter, proposed by Dong et al. (2014), together with Laptop14 and Restaurant14 from SemEval (Semantic Evaluation) (Kirange et al., 2014). Each sample was labelled as either positive, neutral, or negative. More details of each data set are displayed in Table 2.

Parameter Setting

In our experiment, the sentence parser provided by the Stanford CoreNLP toolkit was taken to generate a syntax graph of each sentence. With respect to our model, we used the BERT-base sentence encoder (Devlin et al., 2018), which consisted of 12 hidden layers with 768 hidden layer units per layer. The encoder ran under the PyTorch platform. The hidden state of the last layer of pretrained BERT was taken as the model input to generate sentence embeddings. Aiming at enhancing the word representation, a 30-dimensional part-of-speech (POS) embedding and a 30-dimensional position embedding were incorporated into the word embeddings of our sentence (Sun et al., 2019). The Adam optimizer, with an initialized learning rate of 1e-3 or 1e-4, was employed to upgrade the working performance (Kingma & Ba, 2014). The learning rate of BERT was set to 1e-5 or 2e-5, while the model batch

Table 2.
Statistics of datasets

Data Set		Positive	Neutral	Negative
Twitter	Train	1,561	3,127	1,560
	Test	173	346	173
Laptop	Train	994	464	870
	Test	341	169	128
Restaurant	Train	2,164	637	807
	Test	728	196	196

size was 8 or 32. For regularization, the dropout applied to word embedding generation ranged from 0.1 to 0.6, and the L2-regularization coefficient was 1e-5. Our model was trained with 200 epochs whilst each experiment was initialized randomly. The experimental results were averaged over three runs, with all parameters randomly initialized in each run. The sentiment classification accuracy and Macro-F1 were used as the evaluation metrics.

Baselines

The evaluation of the proposed model was performed by using seven state-of-the-art methods as the baselines.

- **BERT (Bidirectional Encoder Representations from Transformer) (Devlin et al., 2018):** The basic BERT model is established based on a bidirectional transformer. With the form of '[CLS] sentence [SEP] aspect [SEP]' as input, BERT can be applied to ABSA.
- **AEN (Attentional Encoder Network) + BERT (Song et al., 2019):** An attentional encoder network is employed to model between context and target, and the pretrained BERT is applied to the task.
- **TD-GAT (Target-Dependent Graph Attention Network) + BERT (Huang & Carley, 2019):** A multilayer graph attention network is devised, within which the sentiment features are propagated from important syntax neighbouring words to the aspect.
- **R-GAT (Relational Graph Attention Network) + BERT (Wang et al., 2020):** A relational graph attention network is proposed to encode the aspect-oriented dependency tree of the sentence.
- **LCFS (Local Context Focus) + BERT (Phan & Ogunbona, 2020):** Context information and syntactical features are modelled jointly for ABSA tasks.
- **SKGCN (Modeling Syntax and Knowledge via Graph Convolutional Network) +BERT (Zhou et al., 2020):** Two GCNs are employed to deal with the syntax graph and the knowledge graph. The sentiment classification is conducted using multihead attention mechanism.
- **BIGCN (Hierarchical Syntactic and Lexical Graph Convolutional Network) (Zhang & Qian, 2020):** By performing convolution over hierarchical syntactic and lexical graphs, different types of dependency information and lexical cooccurrence information in sentences are extracted.
- **DGEDT (Dependency Graph Enhanced Dual-transformer Structure) + BERT (Tang et al., 2020):** Considering the flat representations learnt from Transformer and graph-based representations learnt from the syntax graph, a dependency graph enhanced dual-transformer network is proposed to support mutual reinforcement between both representations.
- **TGCN (Type-aware Graph Convolutional Network) + BERT (Tian et al., 2021):** Based on different dependency types, a type-aware graph convolutional network is developed where attention is used to distinguish the relations and an attentive layer ensemble is taken to learn from different layers of the model.
- **AGCN (Aggregated Graph Convolutional Network) +BERT (Zhao et al., 2022):** To exploit the node features, two aggregator functions are introduced to iteratively update the representation of each node from its local neighbourhood. The node information is aggregated via subdependencies, and the sentiment dependencies between nodes are captured based on attention mechanism.

Results and Analysis

Experimental results are exhibited in Table 3. One can easily see that the proposed model is a competitive alternative in all evaluation settings. According to Table 3, there is a considerable performance gap between our model and the revised-dependency-tree-based methods. Further, with respect to the exploiting of syntactic structure, our model shows the superiority in improving the working performance on most data sets. Besides, for the models leveraging both semantics and syntax, the establishment of semantic graph and syntactic graph in the proposed model still gives a significant rise. Nevertheless, seeing that the proposed model mainly focuses on syntactic information encoding, our model failed to overperform the DGEDT in Twitter. A possible explanation is that the syntactic structure is relatively incomplete for the samples from the data set Twitter. The experimental results demonstrate the effectiveness of our SGRPN model in integrating the syntactic information and the semantic information, leading to a performance improvement of 1.51% and 0.91% in Lap14 and Rest14, respectively.

Ablation Study

In order to determine the importance of different parts in our model, the ablation study was performed to compare the performance of four variants of SGRPN against the basic model (see Table 4).

The ablating of the syntactic information encoder leads to the accuracy drops of 1.0%, 1.1%, and 1.6% on Twitter, Lap14, and Rest14, respectively. In contrast, the comparable decrease results from

Table 3.
Sentiment classification results

Model	Twitter		Lap14		Rest14	
	ACC	F1	ACC	F1	ACC	F1
BERT	73.27	71.52	77.59	73.28	84.11	76.68
AEN+BERT	75.14	74.15	76.96	73.67	84.29	77.22
TD-GAT+BERT	–	–	80.10	–	83.00	–
R-GAT+BERT	76.15	74.88	78.21	74.07	86.60	81.35
LCFS+BERT	–	–	80.52	77.13	86.71	80.31
SKGCN+BERT	75.00	73.01	79.00	75.57	83.48	75.19
BIGCN	74.16	73.35	74.59	71.84	81.97	73.48
DGEDT+BERT	77.90	75.40	79.50	75.60	86.30	80.00
TGCN+BERT	76.45	75.25	80.88	77.03	86.16	79.95
AGCN+BERT	75.43	74.11	79.94	76.52	82.77	73.29
SGRPN+BERT	**76.70**	**74.96**	**81.01**	**77.99**	**87.21**	**80.98**

Table 4.
Ablation study results

Ablation*	Twitter		Lap14		Rest14	
	ACC	F1	ACC	F1	ACC	F1
$SGRPN\ w/o\ H_{str}$	75.71	73.48	79.91	75.39	85.60	79.01
$SGRPN\ w/o\ H_{sem}$	75.08	73.36	79.43	74.60	86.22	80.56
$SGRPN\ w/o\ Prop$	74.23	72.78	79.74	75.31	86.50	80.84
$SGRPN\ w/o\ Induct$	74.14	72.31	79.43	74.93	84.70	78.12
SGRPN	**76.70**	**74.96**	**81.01**	**77.99**	**87.21**	**80.98**

Note: '$w/o H_{str}$': remove structure module; '$w/o H_{sem}$': remove semantic module; '$w/o Prop$': remove information propagation module; and '$w/o Induct$': remove structure induction module.

the removal of the structure induction unit. The main reason is that both the retention and induction of sentence syntax benefit sentiment information delivery. Moreover, the contribution of the semantic encoding module has a major effect on Twitter and Lap14 but a minor effect on Rest14 because of the lack of syntax in samples from Twitter and Lap14. As compared with the other three modules, the withdrawal of the information propagation module also causes a marginal decline, which indicates the effectiveness of interaction between the syntax and the semantics.

Case Study

The effectiveness of the proposed model was further investigated by the visualization of two cases. The two aspects 'Décor' and 'service' are in the first sentence, 'Décor is nice though service can be spotty.' The relevance values of different words toward the aspect were calculated and presented in Figure 5. The darker the colour, the higher the relevance weight obtained. For the syntax-based method, the major focus is designated to the word 'nice' for sentiment prediction, but the actual opinion word 'spotty' is ignored, so is the semantic-based method. By contrast, our model was capable of assigning a higher weight to 'spotty,' as well as weakening the contribution of 'nice.' The semantic information and the syntactic information were integrated in the proposed model, based on which the sentiment word can be precisely captured through the syntactic neighbouring word 'though.'

For the second example, the efficacy of the structure induction module was verified. In this sentence, the sentiment polarities of aspects 'menu' and 'dishes' are negative and positive, respectively. Based on the sentence parsing, the dependency between 'limited' and 'excellent' was built while that of 'limited' and 'but' was also formulated. In this way, the noise was incorporated into the generation of aspect representation. For our model, the relationship between the aspect and its context words can be established in a more accurate manner. According to Figure 6, the dependencies of aspect word 'dishes' with the long-distance context word 'but' in addition to the opinion word 'excellent' were extracted. As such, our model had its distinctiveness in setting up the connection between aspects and context words, together with eliminating the noise.

Effect of Hyperparameters

In our model, the two hyperparameters α and β were introduced to efficiently integrate the semantic information and the syntactic information. Experiments were carried out on three data sets to clarify the setting of the hyperparameters (see Figure 7). The values of α and β ranged within the interval of [0,1], with the step of 0.1. Based on the adaptive learning via neural networks, the optimal configurations of α and β for Rest14, Lap14, and Twitter were 0.2 and 0.5, 0.3 and 0.4, as well as 0.3 and 0.9, respectively.

Figure 5.
Word relevance scores of removing semantic module, removing structure module, and original model

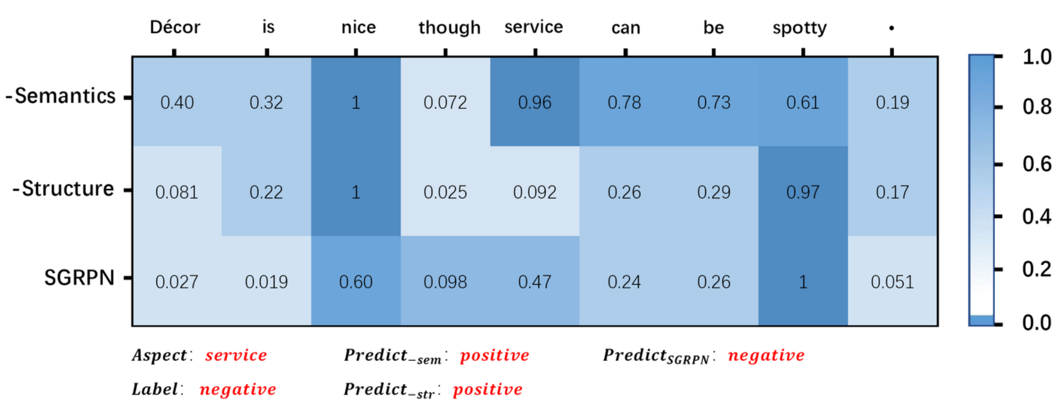
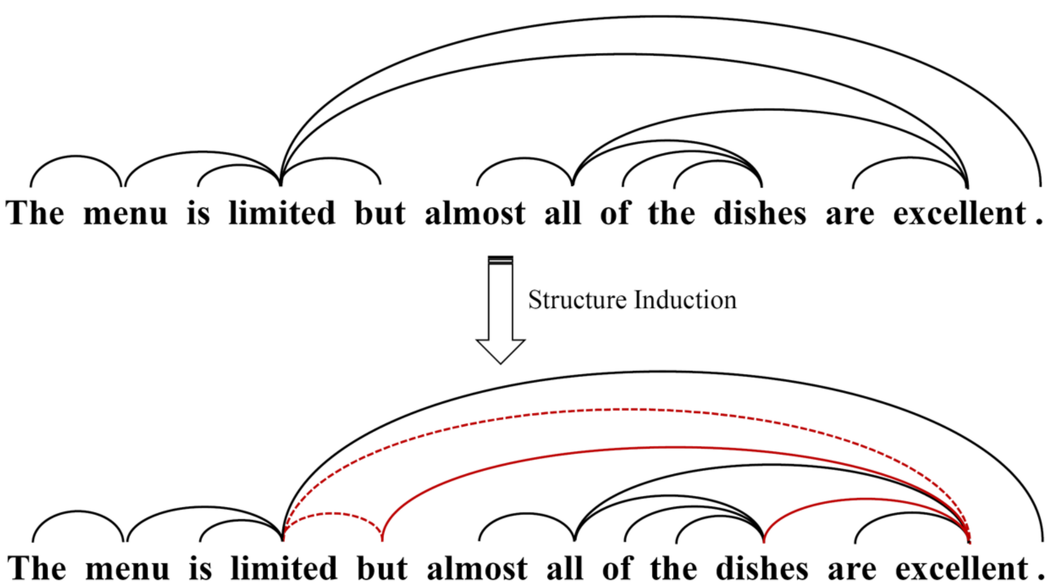


Figure 6.
Sentence syntactic structure with and without structure induction

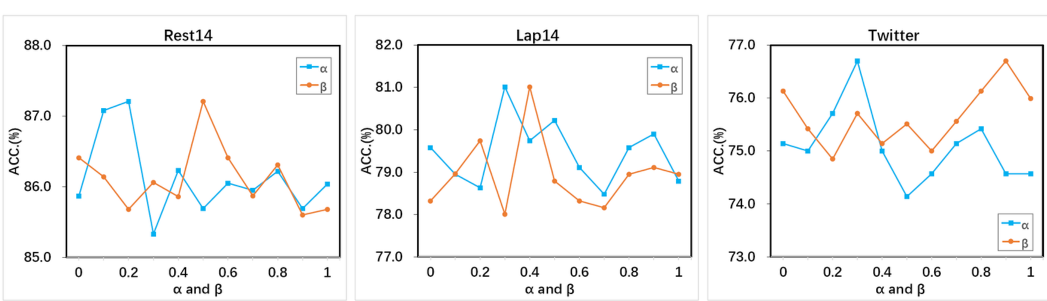


Notably, for $\alpha = \beta = 0$, both the aspect representation and the context representation were solely involved with the semantics without syntactic information. By contrast, when $\alpha = \beta = 1$, both representations consisted of merely the structure induction module outcomes. Taking the result of Rest14 as an example, the best performance was achieved with $\alpha = 0.2$ and $\beta = 0.5$, but the accuracy dropped in line with the increase of α and β. We shall thus propose that the self-adjustment can be performed corresponding to the distinguishing occupation of semantics and syntax in each data set. In this way, the adjusting of the hyperparameters makes a substantial contribution to the information fusion.

CONCLUSION

In this work, a method that integrates sentence syntax and semantics for ABSA tasks is proposed. On the one hand, the syntactic structure of the sentence is dedicatedly exploited and refined, based

Figure 7.
Effect of α and β on Restaurant14, Laptop14, and Twitter using SGRPN model



(a) Restaurant14 Data set (b) Laptop14 Data set (c) Twitter Data set on which a syntactic graph is established and encoded via a GCN. On the other hand, the semantic information is also extracted and encoded using graph convolution. Considering the interaction between the syntax and the semantics, a graph information propagating module is devised for the generation of representations. Experimental results reveal that the proposed method can precisely capture the dependencies among words and sufficiently fuse the syntactic information and the semantic information. Compared with the state-of-the-art methods, our model shows its distinctiveness in improving the sentiment classification results. In addition, the ablation experiment and the case studies further validate the importance of each component in our model.

CONFLICT OF INTEREST

The authors of this publication declare there are no competing interests.

FUNDING STATEMENT

This work was supported in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2023A1515011370, the Characteristic Innovation Projects of Guangdong Colleges and Universities (No. 2018KTSCX049), and the Science and Technology Plan Project of Guangzhou under Grant No. 202102080258.

REFERENCES

Chen, C., Teng, Z., & Zhang, Y. (2020). Inducing target-specific latent structures for aspect sentiment classification. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. doi:10.18653/v1/2020.emnlp-main.451

Chen, P., Sun, Z., Bing, L., & Yang, W. (2017). Recurrent attention network on memory for aspect sentiment analysis. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. doi:10.18653/v1/D17-1047

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). *Bert: Pre-training of deep bidirectional transformers for language understanding*. arXiv. 10.48550/arXiv.1810.04805

Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., & Xu, K. (2014). Adaptive recursive neural network for target-dependent Twitter sentiment classification. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (*volume 2*: Short papers)*. doi:10.3115/v1/P14-2009

Huang, B., & Carley, K. M. (2019). *Syntax-aware aspect level sentiment classification with graph attention networks*. arXiv. 10.18653/v1/D19-1549

Kim, Y., Denton, C., Hoang, L., & Rush, A. M. (2017). *Structured attention networks*. arXiv. 10.48550/arXiv.1702.00887

Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization*. arXiv. 10.48550/arXiv.1412.6980

Kipf, T. N., & Welling, M. (2016). *Semi-supervised classification with graph convolutional networks*. arXiv. 10.48550/arXiv.1609.02907

Kirange, D. K., Deshmukh, R. R., & Kirange, M. D. K. (2014). Aspect based sentiment analysis semeval-2014 task 4. *Asian Journal of Computer Science and Information Technology*, *4*, 72–75.

Koo, T., Globerson, A., Carreras Pérez, X., & Collins, M. (2007, June 28-30). *Structured prediction models via the matrix-tree theorem* [Conference session]. In Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), Prague, Czech Republic. https://aclanthology.org/D07-1015

Li, R., Chen, H., Feng, F., Ma, Z., Wang, X., & Hovy, E. (2021). Dual graph convolutional networks for aspect-based sentiment analysis. *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)*. https://aclanthology.org/2021.acl-long.494

Liu, J., & Zhang, Y. (2017). Attention modeling for targeted sentiment. *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. https://aclanthology.org/E17-2091

Liu, Y., Titov, I., & Lapata, M. (2019). Single document summarization as tree induction. *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)*. https://aclanthology.org/N19-1173

Nan, G., Guo, Z., Sekulić, I., & Lu, W. (2020). *Reasoning with latent structure refinement for document-level relation extraction*. arXiv. 10.48550/arXiv.2005.06312

Phan, M. H., & Ogunbona, P. O. (2020, July). Modelling context and syntactical features for aspect-based sentiment analysis. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 3211-3220). doi:10.18653/v1/2020.acl-main.293

Song, Y., Wang, J., Jiang, T., Liu, Z., & Rao, Y. (2019). *Attentional encoder network for targeted sentiment classification*. arXiv. 10.48550/arXiv.1902.09314

Sun, C., Huang, L., & Qiu, X. (2019). *Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence*. arXiv. <ALIGNMENT.qj></ALIGNMENT>10.48550/arXiv.1903.089588

Sun, K., Zhang, R., Mensah, S., Mao, Y., & Liu, X. (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. doi:10.18653/v1/D19-1569

Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2014). Learning sentiment-specific word embedding for Twitter sentiment classification. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://aclanthology.org/P14-1146

Tang, H., Ji, D., Li, C., & Zhou, Q. (2020). Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. https://aclanthology.org/2020.acl-main.588

Tian, Y., Chen, G., & Song, Y. (2021, June). Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. https://aclanthology.org/2021.naacl-main.231

Tutte, W. T. (2001). Graph theory. In G.-C. Rota (Ed.), *Cambridge mathematical library* (Vol. 21). Cambridge University Press.

Wang, K., Shen, W., Yang, Y., Quan, X., & Wang, R. (2020). *Relational graph attention network for aspect-based sentiment analysis*. arXiv. 10.48550/arXiv.2004.12362

Xu, H., Liu, B., Shu, L., & Yu, P. S. (2019). *BERT post-training for review reading comprehension and aspect-based sentiment analysis*. arXiv. 10.48550/arXiv.1904.02232

Xue, W., & Li, T. (2018). *Aspect based sentiment analysis with gated convolutional networks*. arXiv. 10.48550/arXiv.1805.07043

Yan, Z., Pang, S., & Xue, Y. (2021, October 13–17). *Semantic enhanced dual-channel graph communication network for aspect-based sentiment analysis* [Conference session]. In CCF International Conference on Natural Language Processing and Chinese Computing, Qindao, China. doi:10.1007/978-3-030-88480-2_42

Yang, M., Tu, W., Wang, J., Xu, F., & Chen, X. (2017). Attention based LSTM for target dependent sentiment classification. *Proceedings of the AAAI Conference on Artificial Intelligence*. https://ojs.aaai.org/index.php/AAAI/article/view/11061

Zeng, J., Ma, X., & Zhou, K. (2019). Enhancing attention-based LSTM with position context for aspect-level sentiment classification. *IEEE Access : Practical Innovations, Open Solutions*, *7*, 20462–20471. doi:10.1109/ACCESS.2019.2893806

Zhang, C., Li, Q., & Song, D. (2019a). *Aspect-based sentiment classification with aspect-specific graph convolutional networks*. arXiv. 10.48550/arXiv.1909.03477

Zhang, C., Li, Q., & Song, D. (2019b). Syntax-aware aspect-level sentiment classification with proximity-weighted convolution network. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. doi:10.1145/3331184.3331351

Zhang, M., & Qian, T. (2020). Convolution over hierarchical syntactic and lexical graphs for aspect level sentiment analysis. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. https://aclanthology.org/2020.emnlp-main.286

Zhang, Y., Qi, P., & Manning, C. D. (2018). *Graph convolution over pruned dependency trees improves relation extraction*. arXiv. 10.48550/arXiv.1809.10185

Zhao, J., Wang, X., Shi, C., Hu, B., Song, G., & Ye, Y. (2021). Heterogeneous graph structure learning for graph neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*. doi:10.1609/aaai.v35i5.16600

Zhao, M., Yang, J., Zhang, J., & Wang, S. (2022). Aggregated graph convolutional networks for aspect-based sentiment classification. *Information Sciences*, *600*, 73–93. doi:10.1016/j.ins.2022.03.082

Zhou, J., Huang, J. X., Hu, Q. V., & He, L. (2020). SK-GCN: Modeling syntax and knowledge via graph convolutional network for aspect-level sentiment classification. *Knowledge-Based Systems*, *205*, 106292. doi:10.1016/j.knosys.2020.106292

Weihao Huang is a graduate student in the School of Electronics and Information Engineering of South China Normal University. He is currently studying for a master's degree. His main research fields include natural language processing, text sentiment analysis and deep neural networks.

Shaohua Cai is senior experimenter/chairman of department trade union, Master of Science of Jinan University, majors in computer science, educational technology and e-commerce.

Haoran Li is a master's student in the School of Electronics and Information Engineering of South China Normal University. Her main research interest is natural language processing.

Qianhua Cai received her master's degree in the school of computer science of South China Normal University. After graduation, she joined the School of Electronics and Information Engineering of South China Normal University. Research interests: data mining, text data analysis.