INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 18, Issue: 5, Month: October, Year: 2023
Article Number: 5089, https://doi.org/10.15837/ijccc.2023.5.5089

CCC Publications

AGORA UNIVERSITY PRESS

SenticGAT: Sentiment Knowledge Enhanced Graph Attention Network for Multi-view Feature Representation in Aspect-based Sentiment Analysis

Bin Yang, Haoling Li, Ying Xing

Bin Yang
China Unicom Research Institute
Beijing, 100048, China
researcher_yang@outlook.com

Haoling Li
School of Electronic Engineering
Beijing University of Posts and Telecommunications
Beijing, 100876, China
lihaoling@bupt.edu.cn

Ying Xing*
Beijing University of Posts and Telecommunications
Beijing, 100876, China
*Corresponding author: xingying@bupt.edu.cn

Abstract

Currently, computational intelligence methods, especially artificial neural networks, are increasingly applied to many scenarios. We mainly attempt to explore the task of fine-grained sentiment classification of review data through computational intelligence methods, especially artificial neural networks, and this task is also known as aspect-based sentiment analysis (ABSA). We propose a new technique called SenticGAT which is a multi-view features fusion model enhanced by an external sentiment database. We encode the external sentiment information into the syntactic dependency tree to obtain an enhanced graph with rich sentiment representation. Then we obtain multi-view features including semantics, syntactic, and sentiment features through GAT based on the enhanced graph by external knowledge. We also design a new strategy for fusing multi-view features using the feature parallel frame and convolution method. Eventually, the sentiment polarity of a specific aspect is determined based on the completely fused multi-view features. Experimental results on four public benchmark datasets demonstrate that our method is effective and sound. And it performs superiorly to existing approaches in fusion multiple-view features.

Keywords: Computational Intelligence, Aspect-based Sentiment Analysis, Graph Attention Network, Feature Fusion, Attention Mechanism.

1 Introduction

Our work focuses on trying to identify sentiment tendencies based on user review data through computational intelligence methods, especially artificial neural networks and deep learning [13, 24]. In order to accurately classify the sentiment polarity, we explored a fine-grained sentiment classification task, namely aspect-based sentiment analysis (ABSA) [26]. ABSA is a fine-grained task for sentiment analysis based on aspects. In the ABSA task, sentiment analysis determines the sentiment polarities of multiple aspects, rather than only identifying a single sentiment polarity for the entire sentence. Since the relationship between context words and a particular aspect have a significant influence on the determining of the sentiment polarity of the aspect, the sentiment polarity for different aspects in a sentence is typically distinct in the ABSA task. For example, someone commented on a PC and said, "USB3 Peripherals are significantly slower than the ThunderBolt ones." The sentiment polarities for two aspects USB3 Peripherals and ThunderBolt are negative and positive, respectively.

Earlier research mainly used rule-based or insights approaches. That is easy to understand, however, it bears obvious characteristics of dedicated handcrafted effort and poor performance[6]. Machine learning and deep neural networks have recently been widely used in ABSA tasks to effectively improve model performance [29]. The common practice is to learn a model that can recognize successive features with critical information-related aspect sentiment polarity through machine learning or deep neural network. There are two research branches that have attracted scholars' attention, namely the attention mechanism and the Graph Neural Network (GNN) related approaches in previous works. These two innovations are frequently used in ASBA tasks. Specifically, the practice of combining long short-term memory (LSTM) and attention mechanisms is particularly prevalent[30]. This is mainly because LSTM can extract the sequence elements of the sentence and the attention mechanism may dedicate stronger focus to significant information of context words for a specific aspect. That's beneficial for judging the sentiment polarity of the aspect. In addition, there is a new trend in recent research work that begins to attempt to extract syntactic dependencies in sentences as features based on GNN techniques in ABSA tasks. At present, the syntactic dependencies of aspect and context words are mainly modeled through Graph Convolutional Network (GCN) in ABSA tasks[37].

To sum up, there are the following limitations in the current ABSA research field based on deep neural network methods: Firstly, there are some drawbacks when acquiring features based on GCN. Previous work did not take into account that the contribution of different neighbor nodes to the output results is distinct. In addition, enhancing the model effect with the help of external sentimental knowledge was ignored by the current feature extraction methods On the other hand, at present, most methods only obtain semantic or syntactic features from a single view when constructing aspect features, and disregard the effect of features from different views, such as semantics, syntax and prior sentiment knowledge. Moreover, only a few studies traditionally employ direct concatenation or summation, which are too simple to integrate completely the information of multiple-view features.

Therefore, to address these problems, we propose a new technique called SenticGAT which is a multi-view features fusion model enhanced by an external sentiment database. SenticNet is an available external sentiment knowledge base, which has outstanding performance for reinforcing sentiment representation as a flexible sentimental knowledge source. Consequently, we encode the external sentiment information into the syntactic dependency tree to obtain indirectly a graph with rich sentimental representation information. And we feed the graph into the graph attention network which has the advantage of taking into account contributions from different nodes in a graph to calculate aspect-related sentimental features. For the second limitation, considering the poor representation ability of a single-view feature, we designed a novel multi-view features fusion framework. The framework integrates semantic, syntactic, and external sentiment knowledge, forming a better representation ability than the single-view feature. In addition, we design a new strategy for fusing multi-view features based on the features of a parallel frame and convolution method. Because these two techniques can integrate features from different perspectives very effectively and obtain more advanced representations, better results are obtained than traditional methods in this manner.

To capture the syntactic information of the sentences, we first construct a primitive graph based on the syntactic dependency tree for each sentence. Moreover, we incorporate external sentiment knowledge into the graph in order to encode the sentimental description information into the relation representation. Each sentence could be characterized as a sentiment-enhanced graph. Then the graph generated from the syntactic dependency tree and the sentiment-enhanced graph are both fed into the GAT-based model to generate the syntactic feature and sentiment feature. Therefore, in addition to both features, we obtained semantic features based on LSTM and attention mechanism, so we obtained multi-view features. Ultimately, the sentiment polarity of an aspect is captured through our proposed fusion model based on parallel frame and convolution methods while the multi-view features serve as input to the fusion model for determining the sentiment polarity of a specific aspect.

The main contributions of our work can be summarized as follows:

- We propose a novel graph attention network method senticGAT that introduces external sentimental knowledge to achieve the goal of combining the sentiment and syntactic features. We encode the external sentiment information into the syntactic dependency tree to indirectly obtain a graph with rich sentimental information. And we feed the graph into the graph attention network to calculate aspect-related sentimental features.

- We propose a new method for modeling the relation of context and aspect words in a sentence from multiple-view features including semantics, syntax, and sentiment prior sentiment. In addition, we design a new strategy for fusing multi-view features based on parallel frame and convolution methods. And Better results than traditional methods are obtained in this manner.

- The experimental results on four benchmark datasets fully demonstrate the importance of introducing external sentimental knowledge and also show that our proposed model can effectively fusion multi-view features in the ABSA task.

The rest of this paper is organized as follows. In section 2, we briefly depict the related works. We present our model SenticGAT in section 3 and show the experimental results and analysis in section 4. At last, we conclude this work in section 5.

2 Related work

2.1 Aspect based sentiment analysis

In early studies, researchers mainly extracted features of the specific aspect through rule-based and statistical methods. Most of these methods are manual and labor-intensive [12]. With the emergence of deep learning (DL), many scholars have turned to exploring the ABSA task by introducing neural networks. In the beginning, researchers only extracted the semantics of the text through LSTM, not considering the relationship between aspect and context words[29].

Because of the excellent performance of the attention mechanism in the natural language processing(NLP) and computer vision (CV) tasks, lots of works have tried to enhance the information of context words related aspect through the attention mechanism in ABSA task[30]. While enjoying the benefit of the attention mechanism, researchers also considered integrating other methods to improve the model effect of the ABSA task, such as feature interaction and regularization. [22] utilized the attention encoder network to model the relationship between the context and the aspect, and introduced label smoothing regularization to improve results. These methods simply extract the semantic information from aspect and context words. However, they lack information about the syntactic dependencies. This probably causes that irrelevant context will be involved in determining the sentiment polarity of the aspect.

2.2 Graph Neural Network

There are some works that attempt to introduce Graph Convolutional Network(GCN) [14] into sentiment analysis. [37] builds a GCN on a syntactic dependency tree to solve related syntactic constraints and long-distance word dependence. [31] combines attention mechanism and GCN to capture the syntactic dependence between different aspects of a sentence. [4] proposes a directed GCN to integrate syntactic information extraction and sentiment analysis.

GAT [28] is a novel graph neural network architecture that operates on graph-structured data, leveraging masked self-attention layers. It overcomes the shortcomings of constant edge weights during fusion compared to prior methods based on graph convolutions or their approximations[27]. Recently, [10] primarily uses inter-word dependencies to propose target-dependent GAT on the ABSA tasks. [11] applies the weight given to the relation head to the reshaped dependency tree. The graph-based approach has dominance only in syntactic information extraction. Therefore GAT will inevitably lack semantic information as well as sentiment knowledge during processing. Our approach solves these problems by knowledge embedding and information fusion.

2.3 Enhanced external sentiment knowledge

It has been noted that sentiment knowledge and common sense provide supplementary benefits in natural language processing [23]. Many fields already have integrated the knowledge into deep neural networks.[9] Similarly, tasks involving sentiment analysis also use external sentiment common sense information [21].

SenticNet is a conceptual-level knowledge base or a multidisciplinary linguistic framework that focuses on sentiment analysis at the conceptual level. For tasks such as polarity detection and sentiment recognition through semantics and linguistics, SenticNet does not rely solely on word co-occurrence frequency. [7]. SenticNet has outstanding performance for reinforcing sentiment representation as a flexible sentimental knowledge base. [17] used SenticNet as a foundation and the LSTM model to add common sense to extract aspect-level and context-level sentiment variables from targeted ABSA. [33] showed that SenticNet outperformed other sentiment lexicons by a wide margin. SenticGCN integrates SenticNet into graph convolutional networks and demonstrates the effectiveness in ABSA tasks [16].

3 Methodology

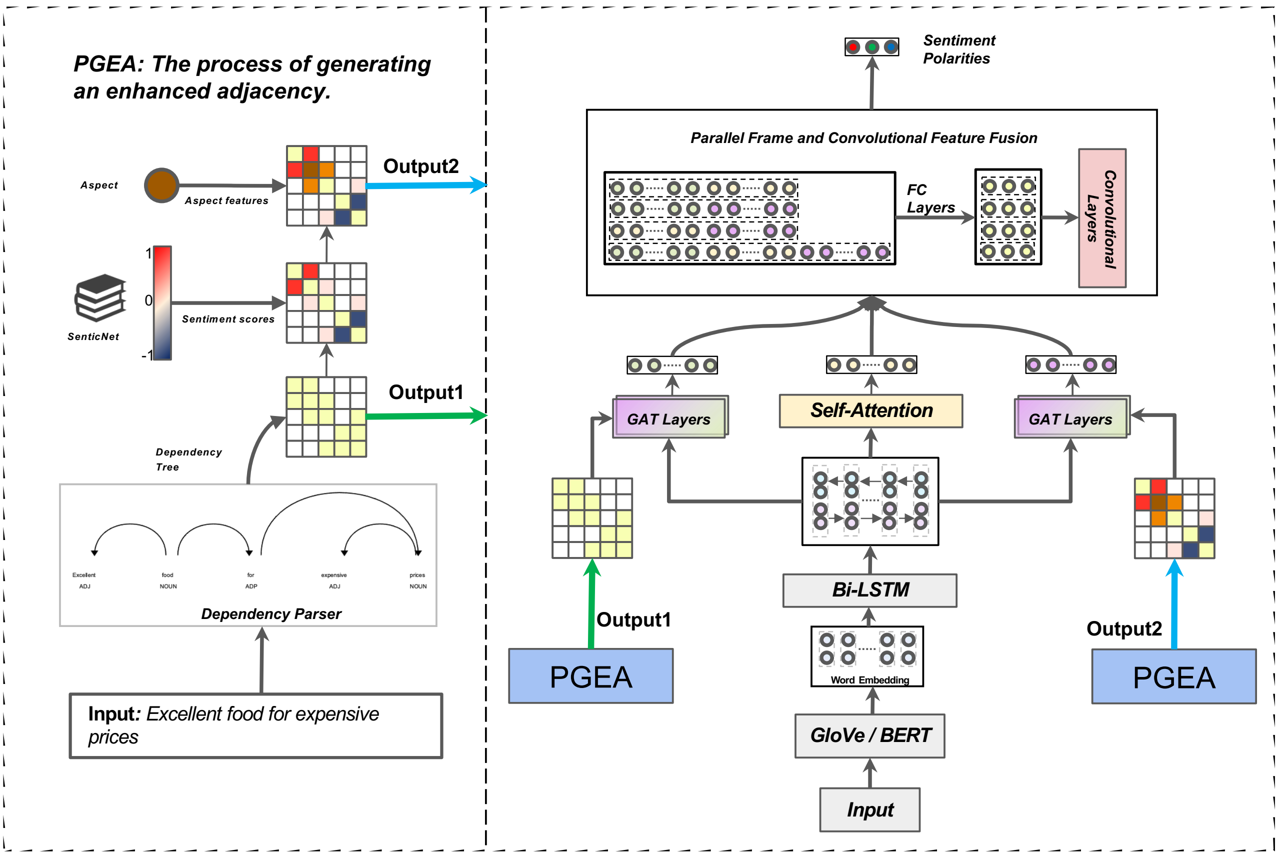

Figure 1: The overall architecture of the SenticGAT

The architecture of our developed SenticGAT model is shown in Figure 1. Firstly, the proposed model takes a sentence as input and parses it syntactically to generate a syntactic dependency tree and construct a graph. Then, the model embeds the sentence using a pre-trained model and passes through a bi-directional LSTM layer. Following that, the sentiment polarities are analyzed in three separate and multi-view branches, i.e., semantic, syntactic and sentiment branches. All features in the sentence are represented exhaustively by this multi-view feature representation. Finally, we collaboratively fused three branches to achieve sentiment classification.

3.1 Embedding

Firstly, $s = \{w_1^s, w_2^s, \ldots, w_\tau^s, w_{(\tau+1)}^s, \ldots, .w_{(\tau+m)}^s, \ldots, w_n^s\}$ is a sentence with n words including a corresponding m words aspect with tokens from $(\tau+1)$ to $(\tau+m)$ [1]. Through word embedding, we map each word to a low-dimensional real-valued vector space. The embedding matrix is denoted as $E \in R^{d\times|V|}$, where d is the embedding dimension of word vectors, and $|V|$ is the size of the vocabulary. In our work, pre-trained word embedding GloVe and BERT are used to initialize word embedding. After individual word embedding, we obtain n-words sentence and m-words aspect embedding vector $s = \{e_1^s, e_2^s, e_3^s, \ldots, e_n^s\}$ and $a = \{e_1^a, e_2^a, e_3^a, \ldots, e_m^a\}$ respectively.

3.2 Semantic features with attention mechanism

The basic idea of the attention mechanism is that each output of the model focuses on only the most important part of the information of the input sequence [36]. In other words, the attention mechanism needs to associate only the most relevant information of the input with the current output, rather than focusing broadly on the entire sentence of the input.

With the word embedding of the sentence, we employ two separate bidirectional LSTMs (Bi-LSTMs) to capture the relativity between the sentence and aspect. It can effectively use the current word and next word of context information, and then summarize the information in two directions to obtain word features. The word embedding vectors including sentence and aspect words are sent into Bi-LSTM layers respectively. A forward $\overrightarrow{\text{LSTM}}$ generates a set of hidden states $\overrightarrow{h}$, and a backward $\overleftarrow{\text{LSTM}}$ generates a set of hidden states $\overleftarrow{h}$. By concatenating together the corresponding forward and backward hidden states, the output hidden states are denoted as $h = [\overrightarrow{h}, \overleftarrow{h}]$. As a result, we obtain the hidden output of Bi-LSTM H_s and H_a for the sentence and aspect respectively as follows:

$$H_s = \{h_1^s, h_2^s, h_3^s, \ldots, h_n^s\} \tag{1a}$$

$$H_a = \{h_1^a, h_2^a, h_3^a, \ldots, h_m^a\} \tag{1b}$$

where $(h_\tau^s, h_\tau^a) \in \mathbb{R}^{2d_h}$ represents the hidden state vector at time step τ from the Bi-LSTM, and d_h is the dimension of a hidden state vector output by an undirectional LSTM.

Multi-head self attention(MHSA) is an attention mechanism that performs multiple scaled dot-product attention in parallel subspaces or heads [22, 27]. Each subspace can pay attention to different feature spaces to improve the performance of the model, and finally put together the results of these subspaces. We define X is the feature representation of input sentences, K, Q, V are the matrices from X by multiplying $W_q \in \mathbb{R}^{d_h\times d_q}, W_k \in \mathbb{R}^{d_h\times d_k}, W_v \in \mathbb{R}^{d_h\times d_v}$, where d_h is dimension of the hidden layer and $d_q = d_k = d_v = \sqrt{d_h}$. Then, an attention function projects Key and Query to an output sequence :

$$\text{Attention(Q, K, V)} = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2}$$

Then, we apply the MHSA procedure to gather scaled-dot attention:

$$S_m = MHSA(X) = \tanh\left(\{H_1 \oplus H_2 \oplus \ldots \oplus H_h\} W_a\right) \tag{3}$$

where $W_a \in \mathbb{R}^{2d_{hid}}$ is the learning weight matrix. H is the output of each attention head, which is obtained by Equation (2). h is the number of attention heads and "$\oplus$" denotes vector concatenation.

[1]We lead shortening and padding for the sentence to obtain the same dimension of word portrayals.

Moreover, we deploy a *tanh* activation function for the result of MHSA. We designate the output obtained from the semantic branch as S_m.

3.3 Syntactic features with Graph Attention Network

In this branch, we attempt to obtain the syntactic features embedded in the sentences. The significance of syntactic features for the comprehension of language models have been shown in the field of NLP, especially in machine translation [1]. Graph Attention Network (GAT) [28] is a variation of graph neural network that utilizes the attention mechanism to encode graph structured information.

In the first place, we build the syntactic dependency tree[2] of the sentences with n words and get the adjacency matrix $\mathcal{A} \in \mathbb{R}^{n\times n}$. Here, we consider the graph addressed by the sentence as an undirected graph based on the fact that the GAT doesn't consider the direction of the graph in the computation cycle. Likewise, this is conducive to the symmetry of the adjacency matrix, which provides convenience for further processing in the following text.

A GAT takes a bunch of words embedding or features, i.e., the output of an LSTM hidden states H and the adjacency matrix $\mathcal{A}$ as inputs. In general, given a word w_i and its neighbor word $w_j \in \mathcal{N}_i$[3], feature aggregation can be done iteratively by calculating the weights through the attention function. We let e_{ij} be computed as a result of an attention function, $f : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$, which computes the attention coefficients of word w_i and neighbor w_j, based on their features:

$$e_{ij}^{l-1} = f\left(H_i^{l-1}, H_j^{l-1}\right) \tag{4}$$

l denotes the number of layers. We inject the graph structure by only allowing node i to interact with nodes in its neighborhood, $j \in \mathcal{N}_i$. These coefficients are then typically normalized using the $softmax$ function. The weight α_{ij}^l indicates to what extent H_i^l depends on H_j^{l-1}:

$$\alpha_{ij}^l = \frac{\exp\left(e_{ij}^{l-1}\right)}{\sum_{k\in\mathcal{N}_i}\exp\left(e_{ik}^{l-1}\right)} \tag{5}$$

Our framework is insensitive to the choice of attention function f: in our experiments, we employed a simple two layers neural network. The parameters of the mechanism are trained jointly with the rest of the network in an end-to-end fashion. To prevent overfitting and make the attention learning process more robust, we use a multi-head attention mechanism[27]. The operations of the layer are independently duplicated K times (each duplication with different parameters), and outputs are feature-wise aggregated:

$$H_i^l = \|_{k=1}^K \sigma\left(\sum_{j\in\mathcal{N}_i} \alpha_{ij}^k \mathbf{W}^k H_j^{l-1}\right) \tag{6}$$

where $\|$ denotes vector concatenation, α_{ij}^k is the attention coefficient derived by the k-th duplication, and W^k is the weight matrix specifying the linear transformation of the k-th replica. σ represents the sigmoid activation function.

To obtain focusing aspect features, we designed a masking mechanism that filters out non-aspect words and keeps solely aspect-specific features:

$$\begin{cases} H_t^l = 0, & 1 \le t < \tau+1, \tau+m < t \le n \\ H_{\text{mask}}^l = \{0, \ldots, H_{\tau+1}, \ldots, H_{\tau+m}, \ldots, 0\} \end{cases} \tag{7}$$

Through graph attention networks over dependency trees, these features H_{mask}^l represent the syntactic information filtered through the mask mechanism. The information after the mask is focusing on the aspect, due to which syntactic information is computed from an entire sentence. In order to simplify the calculation, we utilize dot-product attention and finally obtain the output from the syntactic branch as S_y.

[2]In this work, we use the spaCy toolkit to construct dependency tree of the sentence: `https://spacy.io/`

[3]$\mathcal{N}_i$ is obtained through the adjacency matrix $\mathcal{A}$

Word	Sentiment scores
delight	0.827
fantastic	0.87
general	0.08
food	0.054
ostentatious	-0.99
fearful	-0.85

Table 1: Examples of sentiment scores for words from SenticNet 5

3.4 Sentiment features with SenticNet enhancement

There have been many works on ABSA task using external databases, lexicons, and grammars to augment the data for better classification results [17]. Introducing external data or information can not only improve classification, but it is a current trend in large-scale language models [34].

In this paper, we use a sentiment computing resource library named SenticNet[4]. SenticNet contains the sentiment score for each word according to the concept knowledge base. Especially, sentiment scores are pretty close to 1 for those strongly positive polarities, while for strongly negative polarities, sentiment scores are close to -1. In general, SenticNet is outstanding in ABSA tasks, for example, finding a bewildering word and then assessing sentiment characteristics through its sentiment scores. We extract 39,891 words from SenticNet 5 [2], and the following examples of sentiment scores are shown in Table 1.

In the ABSA tasks, previous works utilize graph neural network-related techniques to process the adjacency matrix obtained from the syntactic dependency parser. However, the adjacency matrix only preserves the syntactic dependencies between words. The value of the adjacency matrix is simply 0 or 1, reflecting whether these words are related, not the strength or weight of the relationship. We believe that embedding the sentiment scores in the adjacency matrix is appropriate, then GAT can process the adjacency matrix.

The integration of sentiment scores into the adjacency matrix is an enhancement of the information representation. We initially obtain the dependency graph for sentences by syntactic dependency parser. The adjacency matrix $\mathcal{A} \in \mathbb{R}^{n\times n}$ of a sentence can be derived as follow[5]:

$$\mathcal{A}_{ij} = \mathcal{A}_{ji} = \begin{cases} 1 & \text{if } w_i \text{ and } w_j \text{ contains dependencies} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

The sentiment score contribution of word W_i to word W_j can be expressed as:

$$Net_{ij} = Net_{ji} = \text{SenticNet}(W_i) \tag{9}$$

where $\text{SenticNet}(W_i) \in [-1, 1]$. Note that W_j is only an offspring of W_i. We argue that the adjacency matrix contains syntactic dependency information, accordingly there is a lack of correlation between words that have no subordination relationship.

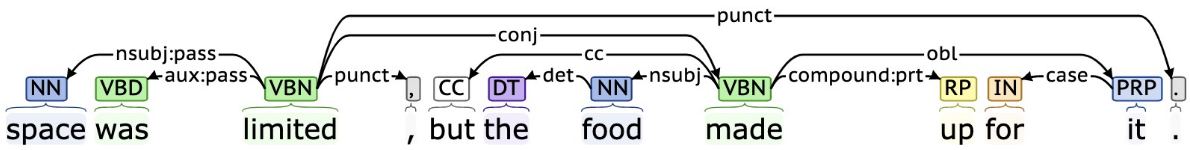

Figure 2: Illustration of the dependency parsing result.

According to the example in Figure 2, there is no subordination relationship between "food" and "limited". If we do not consider this relationship and blindly assign sentiment scores to each other,

[4] http://sentic.net/

[5] Here, as with other calculations based on graph neural networks in ABSA task, we utilize undirected graphs to develop the adjacency matrix.

it will have a negative impact on the classification results. Other methods(e.g. direct utilization of adjacency matrix of dependencies) disregard the subordination between words, resulting in irrelevant words bringing the noise to the aspect words. Additionally, we considered the significant role of the given aspect words in the ABSA task. To strengthen the dependency on aspect and context words, we define aspect representations as follows:

$$Asp_{ij} = Asp_{ji} = \begin{cases} 1 & \text{if } \ w_i \ \text{ or } \ w_j \text{ is an aspect word} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

In English sentences, words may represent different meanings due to the diversity of lexical and syntactic structures of sentences. To avoid semantic bias leading to classification failure, we calculate the semantic similarity based on the cosine similarity.[6] It's a scalar similarity score which is denoted as $Sim_{ij} \in [-1, 1]$ and the larger the value, the more similar it is. From the above reasoning calculations, we denote the sentiment score as:

$$S_{ij} = S_{ji} = \begin{cases} Net + Asp + 1 & i = j \\ (Net + Asp + 1) \times (Sim + 1) & i \neq j \end{cases} \tag{11}$$

The number 1 in the Equation (11) is to ensure that S_{ij} is non-negative. After that, we can obtain the enhanced adjacency matrix $\mathcal{T}$ of the sentence:

$$\mathcal{T}_{ij} = \mathcal{T}_{ji} = S_{ij} \times \mathcal{A}_{ij} \tag{12}$$

We formulated the process of generating an enhanced adjacency matrix for each sentence in Algorithm 1. Finally, the enhanced adjacency matrix will be fed to the GAT for processing as in Section 3.3, and the final output of GAT is noted as S_t.

3.5 Parallel frame and convolutional feature fusion

With Sections 3.2, 3.3 and 3.4, we obtain the multi-views output S_m for the semantic branch, S_y for the syntactic branch, and S_t for the affective branch, respectively. It would be inappropriate to concatenate three branches together straightforwardly. In this subsection, we propose a parallel frame that can coordinate all three branches and complement each other. The purpose of feature fusion is to merge multi-view features into a feature representation that is more discriminative than the original features.

In this fusion model, we firstly concatenate feature representations S_m, S_y and S_t two by two in rows, i.e., $[S_m \oplus S_y]$, $[S_m \oplus S_t]$, $[S_y \oplus S_t]$. Then we fed them into three separate fully connected layers to acquire S_{my}, S_{mt}, and S_{yt} as fused feature information. To guarantee that each branch is independent and to obtain its own local optimum, we do not share parameters for these fully connected layers. In addition, we also concatenate together $[S_m \oplus S_y \oplus S_t]$ in rows and fed it into fully connected layers to obtain S_{myt} as the reference feature information for fusion. To take full advantage of the complement of these multi-view features, we concatenate the fused feature representations together in columns, i.e. $[S_{my}; S_{mt}; S_{yt}; S_{myt}]$. All concatenation processes are performed so that features can take advantage of local information and complement each other. Specifically, the process of fusion highlights the advantages of extracting features from multiple views.

Next, we implement key feature extraction by a convolutional layer. Different from using the convolutional layers in the image task, for the processing of sentence sequences, we use a one-dimensional convolutional kernel $Conv1d$. And the output p of the fusion model can be computed as follow:

$$p = Conv1d([S_{my}; S_{mt}; S_{yt}; S_{myt}]) \tag{13}$$

Through the above parallel input and fusion processes, we obtain multi-view feature representations that contain semantics, syntax and sentiment. In this process, external sentiment knowledge is all around embedded in the context and aspect words to accomplish better sentiment classification results.

[6] We use the spaCy toolkit to calculate semantic similarity: https://spacy.io/api/token#similarity

Algorithm 1: The process of generating an enhanced adjacency matrix for each sentence

Input: Sentence S=$\{w_1^s, w_2^s, \ldots, w_n^s\}$; $\mathcal{A}$: The dependency of the sentence; SenticNet: a set of sentiment scores from SenticNet;

Output: Enhanced adjacency matrix $\mathcal{T}$

$Initialize \mathcal{T} \leftarrow 0$
for $i = 1 \rightarrow n$ **do**
  for $j = i \rightarrow n$ **do**
    while $D(w_i, w_j)$ **do**
      if w_j *is an offspring of* w_i **then**
        $Net_{ij} \leftarrow SenticNet(w_i)$
      else
        $Net_{ij} \leftarrow 0$
      end
      if w_i *or* w_j *is an aspect word* **then**
        $Asp_{ij} \leftarrow 1$
      else
        $Asp_{ij} \leftarrow 0$
      end
      $Sim_{ij} \leftarrow CosineSimilarity(w_i, w_j)$
      if $i = j$ **then**
        $S_{ij} \leftarrow Net_{ij} + Asp_{ij} + 1$
      else
        $S_{ij} \leftarrow (Net_{ij} + Asp_{ij} + 1) \times (Sim_{ij} + 1)$
      end
      $\mathcal{T}_{ij} = \mathcal{T}_{ji} = S_{ij} \times A_{ij}$
    end
  end
end

Finally, we cast the output of the convolution layer as the ultimate sentiment prediction. This model is trained by the standard gradient descent algorithm with the cross-entropy loss and L2-regularization:

$$\mathcal{L} = -\sum_i \sum_j \hat{y}_i^j \log p_i^j + \lambda \|\Theta\|^2 \tag{14}$$

where i indexes the i-th instance of the dataset, and j indexs the j-th sentiment plority. $\hat{y}_i^j$ is the correct distribution of sentiment and p is the predicted distribution of sentence sentiment polarity. Θ represents all trainable parameters, and λ is the coefficient of L2-regularization.

4 Experiments

4.1 Datasets and experimental settings

To comprehensively evaluate the performance of the SenticGAT, we conducted experiments on four datasets(LAP14, REST14, REST15, REST16) derived from SemEval 2014 task 4 [18], SemEval 2015 task 12 [20] and SemEval 2016 task 5 [19]. These four datasets include two categories, laptop and restaurant. Each sample consists of the review sentences, aspects, and the sentiment polarity towards the aspects. These datasets are labeled with three sentiment polarities: positive, neutral and negative. Table 2 shows the number of training and test instances in each dataset.

We extensively validated the effectiveness of our SenticGAT on two pre-trained models, GloVe and BERT. We use them specifically to initialize the word embeddings. The empirical learning rates for GloVe-based and BERT-based SenticGAT are 3e-3 and 5e-5, respectively. To avoid overfitting, we use dropout with a drop rate of 0.6 on the word embeddings. Adam is employed to complete the

Datasets		#Pos	#Neu	#Neg	Total
LAP14	Train	994	464	870	2328
	Test	341	169	128	638
REST14	Train	2164	637	807	3608
	Test	728	196	196	1120
REST15	Train	912	36	256	1204
	Test	326	34	182	542
REST16	Train	1240	69	439	1748
	Test	469	30	117	616

Table 2: Statistics of datasets used in this paper

optimization and training. The experimental results are obtained by averaging 5 times with random initialization, where Accuracy and Macro-Averaged F1 are adopted as the evaluation metrics.

4.2 Comparison models

To thoroughly evaluate the performance of the proposed SenticGAT, we compare SenticGAT with some baselines, which are listed below:

- TNet [15] proposes a method for generating aspect-specific representations of words in a sentence that includes a mechanism for retaining the RNN layer's original contextual information.
- MGAN [8] explores both fine-grained and coarse-grained attention mechanisms to capture context information with BiLSTM, and then learns the interaction between aspect and context words using a multi-grained attention mechanism.
- BERT [5] is the vanilla BERT model, which adopts "[CLS] sentence [SEP] aspect [SEP]" as input.
- ASGCN [37] proposes a graph convolution network (GCN) on the dependency tree of sentences to take use of syntactic information and word dependence.
- BiGCN [38] creates a concept hierarchy on both the lexical and syntactic graphs for the purpose of predicting sentiment.
- kumaGCN [3] combines data from a dependency graph and a latent graph to learn syntactic properties.
- AEGCN [31] proposes the multi-head attention and an improved graph convolutional network built over the dependency tree of a sentence.
- SK-GCN [39] proposes a method for modeling dependency trees and knowledge graphs by syntax-based GCN and knowledge-based GCN together
- A-KVMN [25] proposes an approach that the type information is modeled by key-value memory networks and different dependency results are selectively leveraged.
- attentionGRU [35] proposes an efficient preprocessing scheme with an attention-based GRU model for aspect-based sentiment analysis

4.3 Main results and analysis

As shown in Table 3, the experimental results demonstrate that our proposed SenticGAT model exhibits outstanding results compared to several other models currently in the mainstream. This illustrates the effectiveness of our proposed method on the ABSA task.

Model	LAP14		REST14		REST15		REST16	
	Accuracy	F1	Accuracy	F1	Accuracy	F1	Accuracy	F1
TNet [15]	74.61	70.14	80.42	71.03	78.47	59.47	89.07	70.43
MGAN [8]	75.39	72.47	81.25	71.94	79.36	57.26	87.06	62.29
ASGCN-DT [37]	74.14	69.24	80.86	72.19	79.34	60.78	88.69	66.64
ASGCN-DG [37]	75.55	71.05	80.77	72.02	79.89	61.89	88.99	67.48
BiGCN [38]	74.59	71.84	81.97	73.48	81.16	64.79	88.96	70.84
kumaGCN [3]	76.12	72.42	81.43	73.64	80.69	65.99	89.39	73.19
AEGCN [31]	75.91	71.63	81.04	71.32	79.95	60.87	87.39	68.22
SK-GCN [39]	73.20	69.18	81.04	71.32	80.12	60.70	85.17	68.08
attentionGRU [35]	75.39	70.50	81.37	72.06	80.88	62.48	89.30	66.93
SenticGAT	**76.33**	**72.60**	**82.14**	**74.01**	**82.10**	**66.37**	**90.42**	**73.89**
BERT [5]	77.59	73.82	84.11	76.68	83.48	66.18	90.10	74.16
AEGCN-BERT [31]	78.73	74.22	82.58	73.40	82.71	69.00	89.61	73.93
SK-GCN-BERT [39]	79.00	75.57	83.48	75.19	83.20	66.78	87.19	72.02
A-KVMN-BERT [25]	79.78	76.14	**85.98**	77.94	84.14	68.49	90.52	73.15
SenticGAT-BERT	**80.88**	**77.53**	85.54	**79.73**	**85.24**	**71.23**	**91.07**	**77.04**

Table 3: Model comparison results (%).The best results with each dataset are in bold. The comparison models' results are retrieved from the original papers.

Our model has a significant advantage on four datasets in the GloVe pre-trained case. Specifically, on the REST15 dataset, compared to the current SOTA model BiGCN (*Acc*: 81.16%; $F1$: 64.79%), SenticGAT (*Acc*: 82.10%; $F1$: 66.37%) achieves performance improvements of 0.94% and 1.58% in terms of accuracy and macro-F1 score, respectively. On the REST16 dataset, the experimental performance of model SenticGAT is 1% higher than that of models kumaGCN and attentionGRU in terms of accuracy. Additionally, SenticGAT outperforms attentionGRU by 6.96% with respect to the macro-F1 score. Compared to various approaches based on graph neural networks and attention mechanisms, our model exhibits performance advantages. The experimental results sufficiently demonstrate the ability of enhancing the adjacency matrix with sentiment prior knowledge.

In the BERT pre-trained case, our method achieves notable performance as well. On the LAP14 dataset, compared to A-KVMN-BERT (*Acc*: 79.78%; $F1$: 76.14%), SenticGAT (*Acc*: 80.88%; $F1$: 77.53%) achieves performance improvements of 1.10% and 1.39% in terms of accuracy and macro-F1 score, respectively. On the REST14 dataset, although the performance of SenticGAT is suboptimal in terms of accuracy, it also outperforms all models by at least 1.38% with respect to the F1 score. Moreover, on the REST15 and REST16 datasets, SenticGAT achieves dramatic performance improvements in both the accuracy and F1 score. Experimental results show that our proposed method can significantly improve the performance with the help of the BERT pre-trained model.

Overall, our proposed model integrates the feature information of the sentence from three perspectives (i.e., semantics, syntax, and sentiment knowledge) and utilizes convolutional layers to obtain feature fusion information. It performs considerably better than the previous graph-based models and attention-based models that verify the effectiveness of our innovations.

4.4 Ablation study

Our proposed SenticGAT model is composed of three multi-view feature representation branches and a convolutional feature fusion module. To explore the contribution of each part of the model and to demonstrate the feasibility of our approach, we conduct ablation experiments. The ablation experimental results are shown in Table 4.

Effectivenes of multi-view feature representation branches. We independently ablate the sentiment, semantic, and syntactic feature representation branches of the SenticGAT model. We find that the w/o sentiment experiment has the worst results. On average, the model performance decreases by 2% in both the accuracy and F1 score on the four datasets. The experiment results of w/o sentiment

Model	LAP14		REST14		REST15		REST16	
	Accuracy	F1	Accuracy	F1	Accuracy	F1	Accuracy	F1
SenticGAT	**76.33**	**72.60**	**82.14**	**74.01**	**82.10**	**66.37**	**90.42**	**73.89**
w/o sentiment	73.29	69.81	80.33	72.18	80.45	64.62	88.19	70.20
w/o semantics	75.17	71.92	81.79	72.38	81.55	66.10	89.39	72.33
w/o syntax	75.93	72.01	81.77	73.05	81.37	65.80	89.91	72.84
w/o convolution	75.91	72.13	81.94	73.31	82.02	65.97	89.80	73.17

Table 4: Ablation experiments results (%).The best results with each dataset are in bold.

indicate that the sentiment feature representation branch is the most critical for the model. Since graph relations are established through syntactic dependency trees when constructing the sentiment knowledge enhanced adjacency matrix. The experimental results of w/o syntax are slightly lower than our proposed SenticGAT model. This indicates that the sentiment feature representation branch already contains part of syntactic information, in other words, sentiment knowledge and syntactic dependency are integrated. Furthermore, the experimental results of w/o semantics and w/o syntax are approximately similar. We conjecture that for the ABSA task, sentiment features are more suitable for sentiment polarity prediction compared to semantic features.

Effectivenes of convolutional feature fusion module. To investigate the contribution of convolutional feature fusion to model performance, we design ablation experiments. Specifically, we utilize the fully connected layer to do a linear transformation of the features instead of the convolutional layer to extract features. For a comparison of the entire fusion model framework, we conduct an experimental exploration in Section 4.5. The fusion model improves model performance further, as shown in the w/o fusion experimental results. For the accuracy and F1 score on the four datasets, there is about a 0.5% improvement in model performance. This illustrates that our convolutional feature fusion strategy achieves the complementarity of multi-view features.

4.5 Impact of different fusion modules

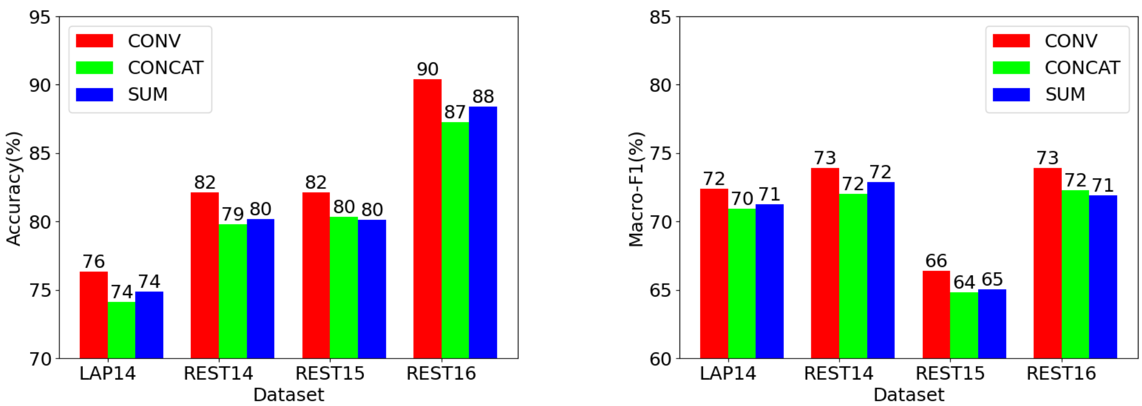

Figure 3: The performance in terms of Accuracy (left) and Macro-F1 (right) for different fusion approaches.

To verify the effectiveness of our proposed convolutional fusion model, we designed comparative experiments, and the experimental results are shown in Figure 3. Specifically, we compare our convolutional fusion model (CONV) with two conventional approaches: three multi-view feature extraction branches are directly concatenated and fused through a fully connected layer (CONCAT); three multi-view feature extraction branches are fed into three independent fully connected layers and fused by summation (SUM). The experimental results show that our proposed fusion model outperforms the other two on all four datasets for both accuracy and Macro-F1 scores. In detail, our proposed fusion model has an approximate 1%-3% advantage over the other methods on all four datasets. This indicates that our fusion model can indeed integrate multiple feature representations and achieve complementarity compared to the two suboptimal methods of direct summation and concatenation.

4.6 Parameter experiment

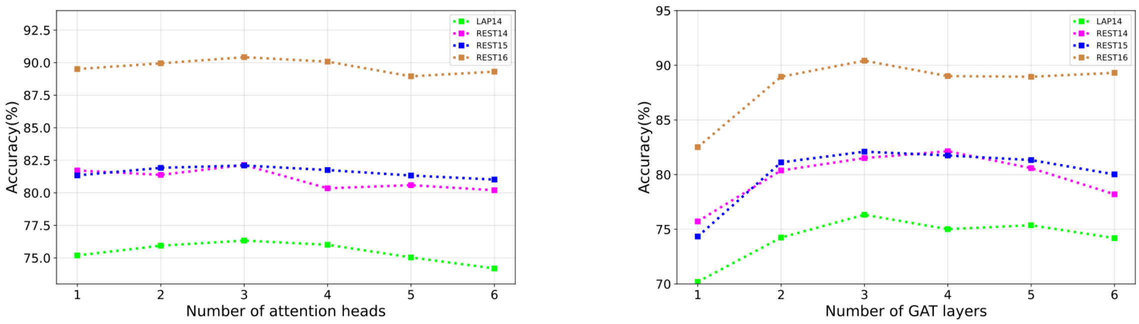

Figure 4: Impact of number of heads in MHA over the four datasets (left). Impact of number of GAT layers over the four datasets (right).

Since the design of the SenticGAT model includes multi-head attention (MHA) and graph attention network (GAT) modules, we explored the effects of attention heads and the number of GAT layers on the model effects on four datasets respectively. We do all experiments on the GloVe pre-trained model and the results are shown in Figure 4.

In Figure 4 left side, we notice that the accuracy fluctuates with the number of attention heads. When the number of attention heads is 3, the model accuracy reaches the optimum. When the number of attention heads continues to increase, the model accuracy starts to decrease. Thus the number of attention heads is set to 3 in our experiments.

According to the experiment results in Figure 4 right side, we found that the model did not work well when there was only one layer of GAT. It suggests that one-layer GCN is insufficient to exploit the sentence's sentiment dependencies with regard to the specified feature. When the number of GAT modules is increased to 2 layers, the model performance improves significantly. The performance of SenticGAT is furthermore improved by adding layers of GAT. On the REST14 data, the best model effect is achieved when the number of GAT layers is 4. For the other three datasets, the best results are achieved when the number of models is 3. Additionally, the performance of SenticGAT varies with the number of GAT layers and effectively declines.

4.7 Case study

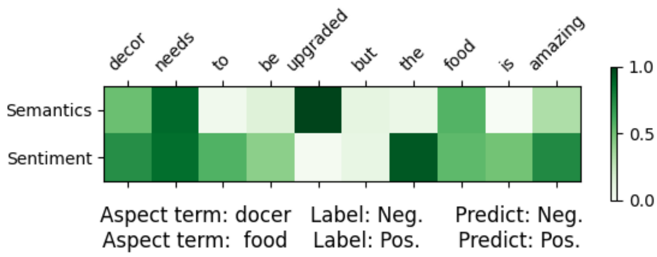

Figure 5: Case study. Visualization of attention scores and sentiment scores in a typical sample from REST14

To qualitatively demonstrate how SenticNet works to enhance the performance of GAT, we present a case study by showing two typical examples. The results are shown in Figure 5, Figure 6. We compared the results of the semantics and sentiment branches, where the attention scores of the semantics

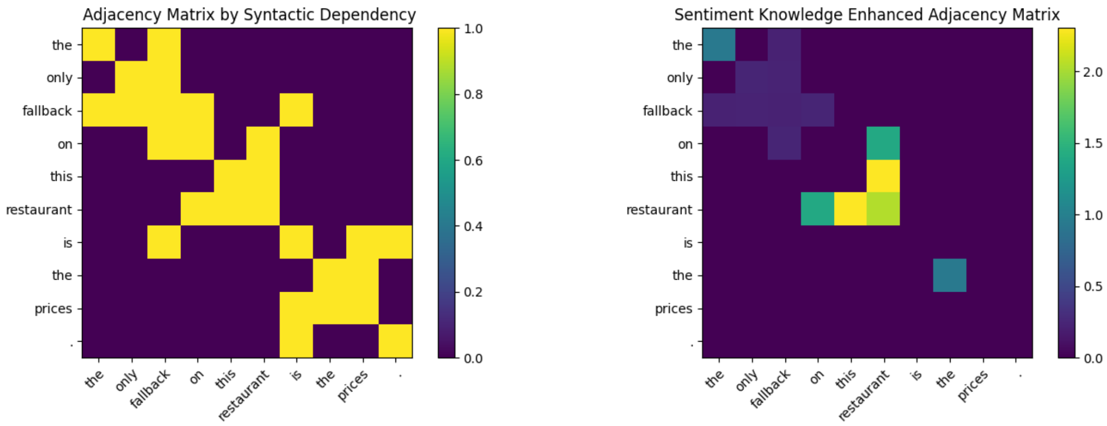

Figure 6: Case study. A typical sample visualization of the adjacency matrix by syntactic dependency (left) and sentiment knowledge enhanced adjacency matrix (right) from REST16. In this case, the aspect is **restaurant**, the label and prediction are the same **nerual**.

branch were derived from the multi-head self attention mechanism and the sentiment results were derived from SenticNet's sentiment scores. Here, the sentiment scores are non-negatively normalized in order to be on the same scale. In addition, smaller values of the sentiment score indicate that the word has a more negative sentiment polarity.

As shown in Figure 5, for the two different aspect terms docer and food, the representation of semantics branch can notice the keywords "upgraded" and "amazing" respectively. The representation of the sentiment branch also focuses on the keyword "amazing". However, extremely low sentiment scores were demonstrated on the word "upgraded", which indicates that the word "upgraded" in this sentence has negative sentiment. This typical sample shows that sentiment knowledge can be complementary to semantics, which facilitates the SenticGAT to understand the sentiment polarity of sentences and make the correct predictions.

On the other hand, in the second instance, we compared the adjacency matrix obtained by parsing syntactic dependencies with the sentiment knowledge-enhanced adjacency matrix. From Figure 6, the values of the adjacency matrix by syntactic dependency are only "0" and "1". In other words, the adjacency matrix only reveals whether the words have dependencies on each other, and without weights. Nonetheless, the sentiment knowledge branch achieves knowledge enhancement by embedding sentiment scores into the adjacency matrix. Thus, the enhanced adjacency matrix not only incorporates the sentiment knowledge but also preserves the dependencies. In addition, we find that the enhanced matrix discards some insignificant dependencies, which is worthwhile for reducing redundant information and decreasing GAT computational consumption. This case confirms our contribution and demonstrates that it is reasonably feasible to embed sentiment scores into the adjacency matrix generated by the syntactic dependency tree.

5 Conclusion

Our paper focuses on the Aspect-based Sentiment analysis task through artificial intelligence and deep learning methods. Traditional methods lack the investigation of sentiment information inherent in the words themselves in the ABSA task and the feature-level integration of semantic, syntactic, and sentiment information. We propose SenticGAT, a sentiment knowledge enhanced graph attention network, to address this issue. To completely extract and supplement the feature information of the aspect words and context, our model starts from three parallel perspectives (i.e., semantics, syntax, and sentiment prior knowledge). We apply the attention mechanism to obtain the semantic representation of aspect words and context. We utilize Graph Attention Network to process the syntactic dependency tree generated via syntactic parsing for obtaining sentence syntactic information. The sentiment prior knowledge is innovatively embedded into the syntactic dependency adjacency matrix and effectively processed by the Graph Attention Network. Finally, to achieve feature complement, we conduct a convolutional fusion of feature information from three parallel perspectives. The experimental results on four datasets LAP14, REST14, REST15 and REST16 show that our proposed SenticGAT is superior to current approaches. In the case of using the GloVe pre-trained model, the experimental results of SenticGAT are optimal compared to those of the baseline model. Additionally, the performance of SenticGAT is significantly improved in terms of F1 scores with the help of the BERT pre-trained model. Our findings contribute towards sentiment analysis to improve quality of life and optimize digital decision-making. Our approach also demonstrates the significant influence of sentiment prior knowledge on the ABSA task. Researchers can explore further studies accordingly to facilitate the flourishing of the sustainable development domain.

Author contributions

Bin Yang: Investigation, Methodology, Project Administration, Data Pre-processing, Writing-Reviewing and Editing.

Haoling Li: Conceptualization, Methodology, Experiment, Software, Visualization, Writing-Original draft preparation.

Ying Xing: Supervision, Validation, Resources, Writing- Reviewing and Editing.

Conflict of interest

The authors declare no conflict of interest.

References

[1] Bastings, J., Titov, I., Aziz, W., Marcheggiani, D., Sima'an, K. (2017). Graph convolutional encoders for syntax-aware neural machine translation, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 1957–1967, 2017.

[2] Cambria, E., Poria, S., Hazarika, D., Kwok, K. (2018). Senticnet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings, in: Proceedings of the AAAI conference on artificial intelligence, 2018.

[3] Chen, C., Teng, Z., Zhang, Y. (2020). Inducing target-specific latent structures for aspect sentiment classification, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 5596–5607, 2020.

[4] Chen, G., Tian, Y., Song, Y. (2020). Joint aspect extraction and sentiment analysis with directional graph convolutional networks, in: Proceedings of the 28th international conference on computational linguistics, pp. 272–279, 2020.

[5] Devlin, J., Chang, M.W., Lee, K., Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018 .

[6] Ding, X., Liu, B., Yu, P.S. (2008). A holistic lexicon-based approach to opinion mining, in: Proceedings of the 2008 international conference on web search and data mining, pp. 231–240, 2008.

[7] Dragoni, M., Poria, S., Cambria, E. (2018). Ontosenticnet: A commonsense ontology for sentiment analysis. IEEE Intelligent Systems 33, 77–85, 2018.

[8] Fan, F., Feng, Y., Zhao, D. (2018). Multi-grained attention network for aspect-level sentiment classification, in: Proceedings of the 2018 conference on empirical methods in natural language processing, pp. 3433–3442, 2018.

[9] Guo, F., He, R., Dang, J., Wang, J. (2020). Working memory-driven neural networks with a novel knowledge enhancement paradigm for implicit discourse relation recognition, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 7822–7829, 2020.

[10] Huang, L., Sun, X., Li, S., Zhang, L., Wang, H. (2020). Syntax-aware graph attention network for aspect-level sentiment classification, in: Proceedings of the 28th international conference on computational linguistics, pp. 799–810, 2020.

[11] Huo, Y., Jiang, D., Sahli, H. (2021). Aspect-based sentiment analysis with weighted relational graph attention network, in: Companion Publication of the 2021 International Conference on Multimodal Interaction, pp. 63–70, 2021.

[12] Jiang, L., Yu, M., Zhou, M., Liu, X., Zhao, T. (2011). Target-dependent twitter sentiment classification, in: Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies, pp. 151–160, 2011.

[13] Kim, E.J., (2022). Applying social computing to analyze the effect of negative emotions on social desirability. Journal of Logistics, Informatics and Service Science 9, 234–257, 2022.

[14] Kipf, T.N., Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907, 2016.

[15] Li, X., Bing, L., Lam, W., Shi, B. (2018). Transformation networks for target-oriented sentiment classification. arXiv preprint arXiv:1805.01086, 2018.

[16] Liang, B., Su, H., Gui, L., Cambria, E., Xu, R. (2022). Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. Knowledge-Based Systems 235, 107643, 2022.

[17] Ma, Y., Peng, H., Cambria, E. (2018). Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm, in: Proceedings of the AAAI conference on artificial intelligence, 2018.

[18] Manandhar, S. (2014). Semeval-2014 task 4: Aspect based sentiment analysis, in: Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014), 2014.

[19] Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., Al-Smadi, M., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O., et al. (2016). Semeval-2016 task 5: Aspect based sentiment analysis, in: International workshop on semantic evaluation, pp. 19–30, 2016.

[20] Pontiki, M., Galanis, D., Papageorgiou, H., Manandhar, S., Androutsopoulos, I. (2015). Semeval-2015 task 12: Aspect based sentiment analysis, in: Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015), pp. 486–495, 2015.

[21] Poria, S., Chaturvedi, I., Cambria, E., Bisio, F. (2016). Sentic lda: Improving on lda with semantic similarity for aspect-based sentiment analysis, in: 2016 international joint conference on neural networks (IJCNN), IEEE. pp. 4465–4473, 2016.

[22] Song, Y., Wang, J., Jiang, T., Liu, Z., Rao, Y. (2019). Attentional encoder network for targeted sentiment classification. arXiv preprint arXiv:1902.09314, 2019.

[23] Song, Y., (2021). Construction of event knowledge graph based on semantic analysis. Tehnički vjesnik 28, 1640–1646, 2021.

[24] Srinivasan, S.M., Shah, P., Surendra, S.S., (2021). An approach to enhance business intelligence and operations by sentimental analysis. Journal of System and Management Sciences 11, 27–40.

[25] Tian, Y., Chen, G., Song, Y. (2021). Enhancing aspect-level sentiment analysis with word dependencies, in: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pp. 3726–3739, 2021.

[26] TRUŞCĂ, M.M., Aldea, A., GRĂDINARU, S.E., ALBU, C., (2021). Post-processing and dimensionality reduction for extreme learning machine in text classification. Economic Computation & Economic Cybernetics Studies & Research 55, 2021.

[27] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems 30, 2017.

[28] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y. (2018). Graph Attention Networks. International Conference on Learning Representations, 2018.

[29] Vo, D.T., Zhang, Y. (2015). Target-dependent twitter sentiment classification with rich automatic features, in: Twenty-fourth international joint conference on artificial intelligence, 2015.

[30] Wang, Y., Huang, M., Zhu, X., Zhao, L. (2016). Attention-based lstm for aspect-level sentiment classification, in: Proceedings of the 2016 conference on empirical methods in natural language processing, pp. 606–615, 2016.

[31] Xiao, L., Hu, X., Chen, Y., Xue, Y., Gu, D., Chen, B., Zhang, T. (2020). Targeted sentiment classification based on attentional encoding and graph convolutional networks. Applied Sciences 10, 957, 2020.

[32] Xiao, Z., Wu, J., Chen, Q., Deng, C. (2021). Bert4gcn: Using bert intermediate layers to augment gcn for aspect-based sentiment classification. arXiv preprint arXiv:2110.00171, 2021.

[33] Xing, F.Z., Pallucchini, F., Cambria, E. (2019). Cognitive-inspired domain adaptation of sentiment lexicons. Information Processing & Management 56, 554–564, 2019.

[34] Xu, P., Patwary, M., Shoeybi, M., Puri, R., Fung, P., Anandkumar, A., Catanzaro, B. (2020). Megatron-cntrl: Controllable story generation with external knowledge using large-scale language models, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 2831–2845, 2020.

[35] Yadav, R.K., Jiao, L., Goodwin, M., Granmo, O.C. (2021). Positionless aspect based sentiment analysis using attention mechanism. Knowledge-Based Systems 226, 107136, 2021.

[36] Zeng, F., Yang, B., Zhao, M., Xing, Y., Ma, Y., (2022). Masanet: Multi-angle self-attention network for semantic segmentation of remote sensing images. Tehnički vjesnik 29, 1567–1575, 2022.

[37] Zhang, C., Li, Q., Song, D. (2019). Aspect-based sentiment classification with aspect-specific graph convolutional networks. arXiv preprint arXiv:1909.03477, 2019.

[38] Zhang, M., Qian, T. (2020). Convolution over hierarchical syntactic and lexical graphs for aspect level sentiment analysis, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 3540–3549, 2020.

[39] 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, 2020.

C O P E
Member since 2012
JM08090

This journal is a member of, and subscribes to the principles of,
the Committee on Publication Ethics (COPE).
https://publicationethics.org/members/international-journal-computers-communications-and-control

Cite this paper as:

Yang, B.; Li, H.; Xing, Y. (2023). SenticGAT: Sentiment Knowledge Enhanced Graph Attention Network for Multi-view Feature Representation in Aspect-based Sentiment Analysis, *International Journal of Computers Communications & Control*, 18(5), 5089, 2023.
https://doi.org/10.15837/ijccc.2023.5.5089