Information Fusion 92 (2023) 289–299

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Full length article

Improving aspect-based sentiment analysis with Knowledge-aware Dependency Graph Network

Haiyan Wu, Chaogeng Huang, Shengchun Deng *

School of Information Management and Artificial Intelligence, Zhejiang University of Finance and Economics, Hangzhou, China

A R T I C L E I N F O

Keywords:
Domain knowledge
Dependency graph network
Sentiment analysis

A B S T R A C T

Aspect-based sentiment analysis (ABSA) aims to mine multiple sentiment–target pairs contained in a review sentence. The main challenge of this task is how to extract the sentiment polarity of a specific sentiment item efficiently. Earlier research focused on recurrent neural networks (RNNs), which implicitly associate the sentiment items with sentiment polarities through an attention mechanism. However, due to the complexity of language and the fact that a sentence contains multiple sentiment pairs, these models often fail to capture sentiment pairs accurately. Most recent efforts have applied syntactic information, especially dependency information, to construct structured models (e.g., tree-based models or graph neural networks) for sentiment analysis. Although these structured models achieve better results, they ignore the domain knowledge related to the entities of the comment sentences. This domain knowledge (e.g., brand reputation, influence) significantly impacts the sentiment polarity. Hence, this paper proposes a **K**nowledge-aware **D**ependency **G**raph **N**etwork (KDGN) based on the dependency graph incorporating domain knowledge, dependency labels, and syntax path. Experimental results on the benchmarking datasets demonstrate that our KDGN significantly outperforms previous state-of-the-art methods on the ABSA task, further illustrating that the domain knowledge, dependency labels, and syntax path are crucial for the ABSA task.

1. Introduction

Sentiment analysis (SA) is a challenging research topic in natural language processing (NLP) [1–3]. With the rapid development of the Internet, digital information resources have also increased explosively. Typically, these sources of information cover a wide range of subjects, such as products, services, government policies, social issues, and comments on blogs, social media, chat forums, and websites. It is essential to analyze the sentiment expressed by these comments effectively. Generally speaking, the sentiments contained in these comment messages are complex and comprehensive. Sentiment analysis falls into three major categories. Document-level sentiment analysis is concerned with determining the sentiment of a specific document. Usually, approval or disapproval, or it can be a five-star rating system, with different stars representing different sentiments [4,5]. Sentence-level sentiment analysis focuses on classifying the sentiment polarity of a given sentence, usually divided into positive, neutral, and negative sentiments, and is the sentence classification task [6]. ABSA task differs from document-and sentence-level sentiment analysis in that it often involves many targets with inconsistent sentiments, such as "This sweater is comfortable to wear, but the color is a bit dark, and the style is a bit old-fashioned" [7–10]. In this statement, "comfortable" to "wear" indicates a positive sentiment towards the sweater, while color and style indicate a negative sentiment. In this case, it is difficult to determine the sentiment of the entire review message because the sentences contain both positive and negative feelings, which requires us to do the ABSA task to solve it. In a review sentence, ABSA can distinguish the sentiment polarity of multiple targets, which can analyze people's sentiment polarities about different attributes or aspects related to the product more subtly than document-level sentiment analysis. However, sentence-level sentiment analysis is suitable for these sentences containing only one pair of the sentiment iterm and sentiment polarity [11,12].

Overall, the ABSA task has passed through three stages. • **Feature Engineering**. Early methods on the ABSA task applied linguistic features to train classifiers and then performed sentiment analysis on customer reviews [2]. Specifically, scholars have filtered nouns or noun phrases pairing with sentiment words sentence by sentence and employed classical machine learning algorithms, e.g., Support Vector Machine (SVM) [13], Logistic Regression [14] and Naive Bayes [15] to perform sentiment analysis and draw some valuable conclusions. These methods are based on high-frequency nouns or high-frequency noun phrases appearing in comment sentences, some of which can

* Corresponding author.
E-mail addresses: wuhy2020@zufe.edu.cn (H. Wu), cghuang@zufe.edu.cn (C. Huang), dengsc@zufe.edu.cn (S. Deng).

https://doi.org/10.1016/j.inffus.2022.12.004
Received 25 January 2022; Received in revised form 30 November 2022; Accepted 3 December 2022
Available online 12 December 2022

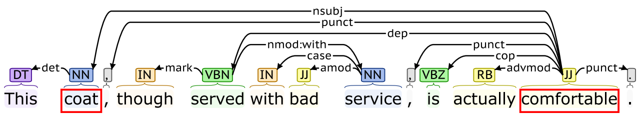

Fig. 1. An example of dependency tree.

reflect the sentiment polarity shown by the reviewer towards the entity or an aspect of the entity. Such approaches are simple and effective in that the sentiment item and sentiment polarity must appear in a sentence [16,17]. However, there are obvious shortcomings in these methods. Usually, sentiment analysis is based on features extracted from high-frequency nouns and noun phrases. In fact, some high-frequency nouns are not sentiment items. For example, in consumer comments, some high-frequency nouns such as "dollars" or "centimeters" are not sentiment items. In addition, there are special aspects that are not often mentioned, which occur less frequently and are ignored [18].

• **Sequence-based Information Modeling**. With the continuous development of neural networks, many works based on neural networks such as Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM) networks and Convolution-based Network have been proposed [3,4,19,20]. Most of these neural network-based approaches consider the sentence as a sequence of words combined, and sentiment polarities between sentiment items and their contexts are modeled by recurrent neural networks [21] to embed them into the sentence representation [9,22]. However, their performance suffers from the inadequacy of RNNs to capture remote dependencies and the loss of syntactic information [8,9].

To overcome the above deficiencies of RNNs-based models, attention mechanisms [23] and Gated Convolutional Networks [24,25] have been used to alleviate the long dependencies of words in sentences [10,11]. These approaches, however, ignore syntactic information. This is not conducive to identifying the sentiment features of the target object with which it is directly related. For example, when sentiment item is separated from its sentiment polarity, it is difficult to capture the associated sentiment word in a sequence [9]. The dependency tree of the example sentence in Fig. 1 manifests that although the sentiment polarity is far away from the sentiment item in the example sentence, their actual dependencies are close to each other. In this case, if the syntactic information is not favorable, the training from the word serialization alone may lead to the combination of the sentiment item (e.g., "service") with the sentiment polarity (e.g.,"comfortable"). Hence, if we integrate dependency information into the model, we can explicitly capture the latent sentiment pairs in the sentences, which is difficult for the serialized models.

• **Structure-based Information Modeling**. Tree-based models and graph neural networks or their variants are the two main types of structured models. **(1) Tree-Based Models.** To exploit syntactic information, some studies proposed using phrase trees or dependency trees as input to build the tree models, which can be divided into explicit tree models like Tree-LSTM [26] and implicit models such as Gumbel Tree-LSTM [27]. Tree-LSTM is a variant of Recursive Neural Networks (RvNN), which extracts syntactic information from the sentence well and introduces new ideas for modeling neural networks. However, the effect of these tree-based models like Tree-LSTM are affected by syntactic tree parsing tools. The pre-parsing results of the implicit tree have many errors, which leads to the limitations of the models built based on it [28]. In contrast to explicit syntactic modeling, implicit syntactic models (e.g., Gumbel Tree-LSTM) leave the tree's structure to be freely learned by the neural network, which applied the Gumbel-Softmax estimator to dynamically determine the parent and child nodes among the candidate nodes. Such methods tend to produce invalid nodes and invalid syntactic structures [29–31]. Then a Recurrent Convolutional Neural Network with gates (grConv) was put forward to employ a gate mechanism to control the information transfer from child to father nodes in a possible syntactic tree [32]. **(2) GCN-Based Models.** Although those tree-based models are effective in avoiding syntactic parsing errors, which often produces some unstable or meaningless syntactic structures [33–35]. In addition, some studies applied the dependency tree of a sentence as input and performed sentiment analysis by constructing different graph neural network models, such as the dependency tree-based graph convolutional neural network [7,36], Attention Graph Convolutional Network (TD-GAT) [9,37], Graph Convolutional Network(GCN) based on the edge information of the dependency tree (ASEGCN) [38], and there are other graph neural networks based on dependency trees or other information fusion [39,40] for ABSA task [41–43]. These graph neural networks apply syntactic information in their modeling and achieve good results, which shows that graph neural networks are so powerful in capturing information [44–47]. In addition, some scholars have done lots of experiments on complex potential emotional polarities such as neutrality and ambivalence, and have achieved significant results [35,48,49]. Recently, with the prevalence of graph neural networks, there has been a breakthrough in capturing neutral and ambivalent sentiments of comment information using graph neural networks aggregating node edge information [50,51].

Although these models make use of edge information, they ignore the dependency labels on the edges, e.g., "nsubj" ("coat", "comfortable") in Fig. 1. In fact, such dependency labels information are helpful for mining sentiment pairs [52]. Besides, these models also ignore domain knowledge impacting sentiment polarity, such as brand reputation, influence, and other information. Therefore, in this paper, we propose a novel **K**nowledge-aware **D**ependency **G**raph **N**etwork (KDGN) for ABSA task. In particular, we construct a knowledge graph based on review information about the brand. To capture sentiment polarity, brand-related knowledge graphs are incorporated into the dependency graphs. Finally, we apply a Relational Graph Attention Network (RGAT) with Knowledge-aware Dependency Graph to aggregate the nodes, edges, syntax path , and dependency labels on the edges and perform on the ABSA task. The main contributions of this paper are summarized as follows:

- We integrate domain knowledge into the dependency tree of review sentence and construct Knowledge-aware Dependency Graph to capture latent sentiment polarity using brand information like brand reputation, influence and other information.
- A Knowledge-aware Dependency Graph Network is proposed to model dependency graph with domain knowledge for the ABSA task. The dependency labels on the edges, domain knowledge and syntax path are employed as modeling the dependency graph network.
- Experimental results on the benchmark datasets (i.e., Restaurant, MAMS) manifest that our KDGN significantly outperforms the state-of-the-art methods on the ABSA task.

2. Related work

Sentiment analysis is becoming very popular in both research and business due to the large amount of opinionated text currently generated by Internet users [53]. Aspect-based sentiment analysis (ABSA), a subtask of sentiment analysis, is a more complex task that includes the identification of sentiments and aspects and has drawn much attention these years. In recent years, most works have focused on integrating attention mechanisms into neural networks to capture aspect–sentiment pairs in comment sentences [10,54,55]. Among them, Wang et al. proposed an attention-based LSTM to identify important sentiment information related to the target aspect [10]. Similarly, He et al. built a conventional attention-based LSTM to capture each context word towards a target by modeling their semantic associations [56]. Tang et al. designed a memory network with multi-hop attention and external memory for the ABSA task [57]. Chen et al. introduced a multilayer attention mechanism to capture long-range the sentiment polarity of sentiment item [55]. Similarly, several other scholars have made similar attempts, all with positive results [58–60]. In many cases, sentiment words usually appear close to sentiment items, suggesting that mining syntactic structure implicitly using attention mechanisms is effective and efficient. Such approaches solve the long dependencies of words problem and the gradient disappearance or explosion problem caused by RNN-based models, offering new idea for the ABSA task [44,61]. Some studies applied language models such as BERT to capture aspect–sentiment pairs to enhance language representation and thus improve the accuracy of ABSA tasks [53,62,63].

Although the attentional mechanism plays an important role in previous state-of-the-art neural models, it is not sufficient to benefit from attention mechanisms to capture aspect–sentiment pairs when the aspect and sentiment words are positioned far apart in a comment sentence. In Fig. 1, the aspect word "coat" and the sentiment word "comfortable" are often confused with the connections. To solve this problem, some scholars have proposed using syntactic information to capture aspect–sentiment pairs, especially the dependence relationship between words. Huang et al. proposed a target-dependent graph attention network (TD-GAT) based on dependency graphs for aspect-level sentiment classification [9]. Zhang et al. constructed a proximity-weighted convolutional network to provide an aspect-specific syntax-aware representation of contexts [64]. Some studies built the Graph Neural Networks based on dependency tree to capture aspect–sentiment pairs, and experimental results showed that the graph neural network based on dependency information can significantly improve the accuracy of the ABSA task [12,65], which indicates that graph neural networks constructed based on dependency information have powerful ability to capture aspect–sentiment pairs [66]. Besides, some studies integrated conventional knowledge (e.g., Sentic GCN integrating sentiment knowledge [50], BiERU with the bidirectional emotional recurrent unit [67], Syntactic-GCN using semantic knowledge [51]) and syntactic information (e.g., SEDC-GCN fusing syntactic knowledge [68]) to construct the Graph Neural Networks for ABSA task [69,70]. And some other scholars have focused on Affective Computing and Sentiment Analysis [71], which play a crucial role in transforming today's AI systems into next-generation emotional AI devices, and it is a highly interdisciplinary field of research that spans Psychology, Cognitive Science and Computer Science. Affective Computing and Sentiment Analysis are concerned with the computation, interpretation [72] or neurosymbolic AI (e.g., SenticNet 7) [73], and generation of human emotions or moods [74,75].

These studies demonstrated that syntactic knowledge, external knowledge (e.g., sentiment knowledge) are also essential for sentiment analysis. In other words, both syntactic information and external knowledge are more critical information for ABSA tasks. Instead, we propose a knowledge-aware dependency graph for sentiment analysis based on a syntactic dependency graph integrating the domain knowledge, dependency labels, syntax path, and then apply R-GAT to simultaneously fuse node and edge information and the dependency labels information on edges. Finally, Ablation Studies demonstrate that our KDGN model incorporating domain knowledge and dependency labels is effective and significantly improves accuracy compared to state-of-the-art models.

3. KDGN: Knowledge-aware Dependency Graph Network

In this section, we construct the Knowledge-aware Dependency Graph Network in three steps according to K-BERT [76,77]. Firstly, the knowledge graph of corresponding words in the comment sentences is constructed, as illustrated in Fig. 2 ("Knowledge Layer"). Secondly, the dependency tree of the comment sentences is built by Stanford CoreNLP [78,79] and Biaffine parser, respectively [80], whose structure is shown in Fig. 2 ("Dependency tree") with CoreNLP. And the knowledge graph constructed with the help of the previous steps fused knowledge into the dependency tree to form the Knowledge-aware Dependency Graph. Finally, we establish the Relational Graph Attention Network as shown in Fig. 4 with the Knowledge-aware Dependency Graph to aggregate the node information, edge information, dependency labels, and domain knowledge, respectively.

3.1. Knowledge layer

We establish the knowledge graph of named entities of the review sentences with YAGO [81–83]. YAGO is a linked database developed by the Max Planck Institute in Germany, integrating data from three primary sources(i.e. Wikipedia, WordNet, and GeoNames) [84]. YAGO contains a rich entity classification system because YAGO integrates WordNet's vocabulary definitions with Wikipedia's classification system, making YAGO have a more prosperous entity classification system. YAGO also considers temporal and spatial knowledge by adding attribute descriptions of temporal and spatial dimensions for many knowledge entries. Therefore, in this paper, we apply YAGO to capture domain knowledge of named entities in review sentences as additional information to mine the potential sentiment polarity of sentiment items. For example, when a user reviews three brands (e.g., Apple, Samsung, and Huawei) of cell phones, he or she has positive sentiment polarity towards the iPhone because she always prefers them. However, another user who likes Huawei mobile phones has a negative or neutral sentiment polarity towards iPhones. To summarize, we first identify the named entities in a comment sentence and then fuse the knowledge information associated with them.

3.2. Knowledge-aware Dependency Graph

Algorithm 1: Knowledge-aware Dependency Graph

Input: dependency tree $T = (V, E_{\text{edge}})$), where $V = \{v_1, \cdots, v_{|V|}\}$, $E_{\text{edge}} = \{(r_{ji}, v_i, v_j)\}$;
Entity list $E_{\text{entity}} = \{e^1_{\text{entity}}, \cdots, e^K_{\text{entity}}\}$
Output: Knowledge-aware Dependency Graph (KDG): $T_{\text{KDG}} = (V_{\text{KDG}}, E^{\text{edge}}_{\text{KDG}})$

```
1 for each entity e^i_entity ∈ E_entity do
2     e^i_KG = K_Query(t, 𝕂);
3     t = K_Inject(t, e^i_KG);
4 end
```

Algorithm 1 demonstrates that how to construct the Knowledge-aware Dependency Graph (KDG). Firstly, The dependency syntactic tree T and entities E_{entity} of the sentences through Stanford CoreNLP and Biaffine parser, respectively, and knowledge base YAGO $\mathbb{K}$ are the inputs of Algorithm 1. Here, the dependency tree T is represented as $T = (V, E_{\text{edge}})$, where $V = \{v_1, \dots, v_{|V|}\}$, $E_{\text{edge}} = \{(r_{ji}, v_i, v_j)\}$. Here we refer to the two operations (i.e., $K\_\text{Query}(t, \mathbb{K})$, $K\_\text{Inject}(t, e^i_{\text{edge}})$) of K-BERT [77] to integrate entity knowledge information into the syntactic tree to form the knowledge-aware dependency graph. In $K\_\text{Query}(t_i, \mathbb{K})$, all the entity names involved in the ith sentence tree t_i are selected to query their corresponding triples from $\mathbb{K}$, where E_{KG} is a collection of triples containing knowledge information, and can be represented as follows:

$$E_{\text{KG}} = \left[(r_{i0}, w_i, w_{i0}), \dots, (r_{ik}, w_i, w_{ik})\right] \tag{1}$$

The $K\_\text{Inject}(t_i, E_{\text{KG}})$ operation injects E_{KG} into the sentence tree t by stitching the triples in E_{KG} to their corresponding positions, and generates the Knowledge-aware Dependency Graph t. An example of Knowledge-aware Dependency Graph Network (KDGN) is shown in Fig. 4.

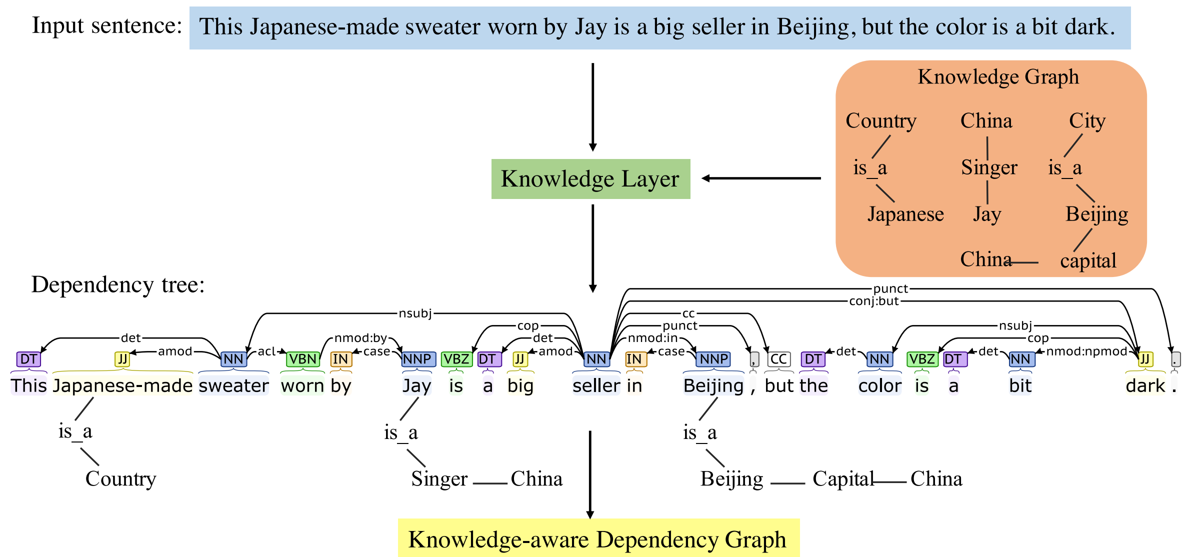

Fig. 2. **An example of Knowledge-aware Dependency Graph**. ***Context Information Representation***: the dependency tree for "This Japanese-made sweater worn by Jay is a big seller in Beijing, but the color is a bit dark.", and the domain knowledge (Knowledge Graph) of "Japanese", "Jay" and "Beijing" with YAGO. ***Information Aggregation Layer***: the knowledge-aware syntactic graph for "This Japanese-made sweater worn by Jay is a big seller in Beijing, but the color is a bit dark", which incorporates domain knowledge into the dependency graph. Finally, sentiment classification is performed by the softmax function.

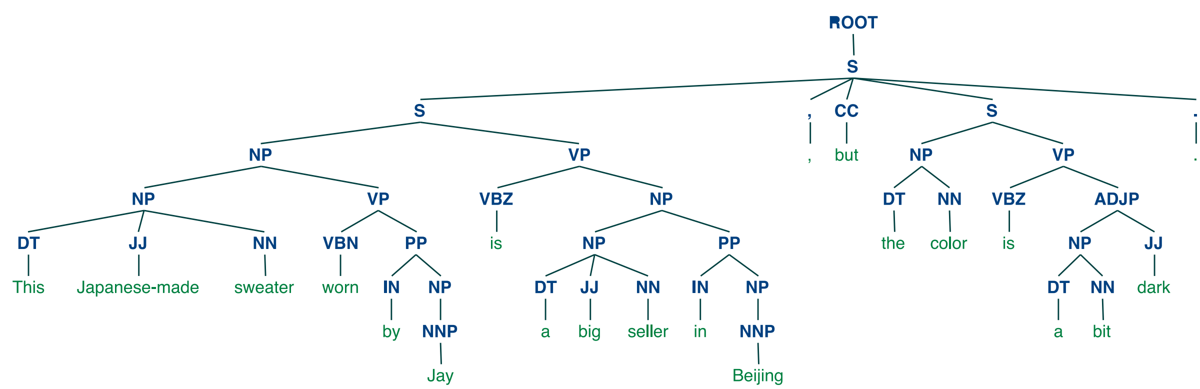

Fig. 3. An example of the phrase tree.

Table 1
Syntax path example.

Words	Syntax path
This	S→S→NP→NP →DT
Japanese-made	S→S→NP→NP →JJ
sweater	S→S→NP→NP →NN
worn	S→S→NP→VP →VBN
by	S→S→NP→VP →PP→IN
Jay	S→S→NP→VP →PP→NP→NNP
…	…
dark	S→S→VP→ADJP →JJ

3.3. Context information representation

Single words do not contain sequence information. In most cases, a word's meaning can be inferred from its context. It is extremely significant to represent words in a specific context [85]. Therefore, we apply a global context bidirectional LSTM (Bi-LSTM) [86] to generate context-enhanced word embeddings and global context vectors. Moreover, inspired by previous work on ABSA [87], we found that syntactic information, especially phrase structure, is very significant in capturing word collocation rule information. For example, in Fig. 3, each word in the phrase tree of the example sentence corresponds to a syntax path [88,89], as shown in Table 1. Therefore, we integrate the syntax path of each word to enhance the representation of words. Suppose the input sentence is $s = \{w_1, \ldots, w_t, \ldots, w_{|s|}\}$, where w_t is the tth word in the sentence and $|s|$ is the sentence length. The word vectors $\mathbf{w}_t \in \mathbb{R}^d$ is initialized with BERT [90], where d is the dimension of word vectors. We employ Bi-LSTM to enrich the word contextual vectors with syntax paths $\mathbf{p}_t \in \mathbb{R}^d$ in the sentences. At position t, the enriched word vector $\mathbf{h}_t$ is:

$$\mathbf{h}_t = \mathbf{w}_t + \mathbf{h}_t^f + \mathbf{h}_t^b + \mathbf{p}_t \tag{2}$$

where $\mathbf{h}_t^f$ and $\mathbf{h}_t^b$ denote that the Bi-LSTM captures past and future context information of the sequence of words $\{w_t\}_{t=1\ldots|s|}$, respectively. $\mathbf{h}_t$ are the inputs of R-GAT modules, the syntactic and global context information are both embedded in $\mathbf{h}_t$, guiding the propagation of information in higher layers.

3.4. Relational Graph Attention Network

Previous studies have demonstrated that *GAT* has better results on ABSA tasks. However, this process does not consider dependencies, and some critical dependency label information may be lost. In *GAT*, each node in the graph can be assigned different weights to its neighboring nodes based on their characteristics. Accordingly, neighboring nodes will impact the relationship between them except for the importance of that node. Their relationships are different, and their weights should

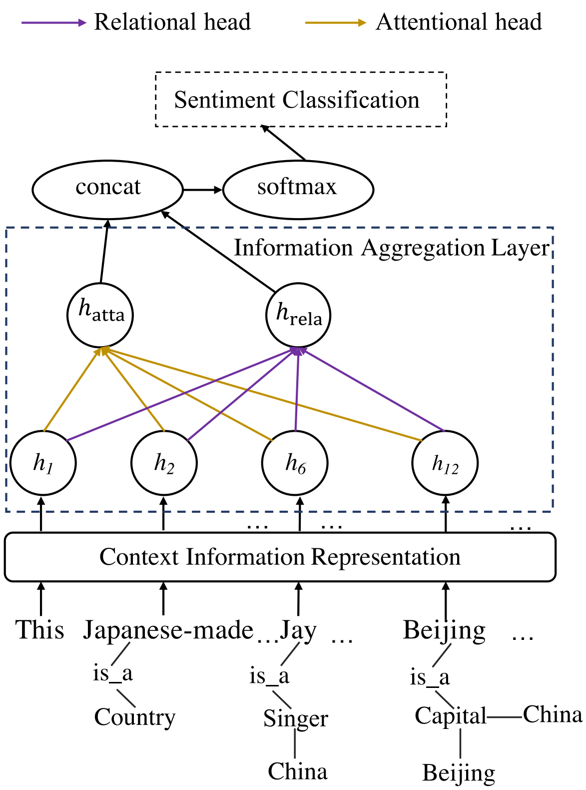

Fig. 4. An example of Relational Graph Attention Layer. ***Knowledge Layer***: the dependency tree for "This Japanese-made sweater worn by Jay is a big seller in Beijing, but the color is a bit dark", and the domain knowledge (Knowledge Graph) of "Japanese", "Jay" and "Beijing" with YAGO. ***Knowledge-aware Dependency Graph***: the knowledge-aware syntactic graph for "This Japanese-made sweater worn by Jay is a big seller in Beijing, but the color is a bit dark", which incorporates domain knowledge into the dependency tree.

be separate and should not be ignored. Intuitively, neighboring nodes with various dependencies should have other impacts [91]. Therefore, in this paper, we propose a relational graph attention network based on the knowledge-aware dependency graph to aggregate the edge information between nodes and their corresponding dependency labels, the structure of which is shown in Fig. 4. *R-GAT* contains K node attention heads and M edge relational heads, where node information aggregation is the same as *GAT* using multi-headed attention to aggregate the representation of neighboring nodes. That is, multiple $\mathbf{W^k}$ are used simultaneously to calculate self-attention, and then the results of each $\mathbf{W^k}$ are combined (concatenated or summed) as follows:

$$\mathbf{h}^{\mathbf{l+1}}_{\mathbf{node_i}} = \|_{k=1}^{K} \sum_{j\in\mathcal{N}_i} \alpha_{ij}^{lk}\mathbf{W}^{\mathbf{l}}_{\mathbf{k}}\mathbf{h}^{\mathbf{l}}_{\mathbf{j}} \tag{3}$$

where $\|$ denotes the connection, α_{ij}^{lk} and $\mathbf{W^k}$ is the calculation results obtained from the k multiple head attention, α_{ij}^{lk} is the normalized attention coefficient calculated from the kth attention of the lth layer. Since $\mathbf{W^k} \in \mathbb{R}^{F'\times F}$, here again $\vec{\mathbf{h}}'_i \in \mathbb{R}^{KF'}$ can be obtained by taking the summation of $\vec{h}'_i$:

$$\vec{\mathbf{h}}'_{\mathbf{node_i}} = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j\in\mathcal{N}_1}\alpha_{ij}^{lk}\mathbf{W}^{lk}\vec{h}_j\right) \tag{4}$$

$$\alpha_{ij}^{lk} = attention(i, j) \tag{5}$$

and the ith relational head information h_{edge} aggregation in the l+1 layer is calculated as follows:

$$\mathbf{h}^{\mathbf{l+1}}_{\mathbf{edge_i}} = \|_{m=1}^{M} \sum_{j\in\mathcal{N}_i} \beta_{ij}^{lm}\mathbf{W}^{\mathbf{l}}_{\mathbf{m}}\mathbf{h}^{\mathbf{l}}_{\mathbf{j}} \tag{6}$$

$$\mathbf{g}^{\mathbf{lm}}_{\mathbf{ij}} = \sigma\left(\text{relu}\left(\mathbf{r_{ij}}\mathbf{W_{m1}} + \mathbf{b_{m1}}\right)\mathbf{W_{m2}} + \mathbf{b_{m2}}\right) \tag{7}$$

$$\beta_{ij}^{lm} = \frac{\exp\left(\mathbf{g}^{\mathbf{lm}}_{\mathbf{ij}}\right)}{\sum_{j=1}^{\mathcal{N}_i}\exp\left(\mathbf{g}^{\mathbf{lm}}_{\mathbf{ij}}\right)} \tag{8}$$

where *R-GAT* contains K node attention heads and M edge relation head attentions. $\mathbf{r_{ij}}$ represents the embedding of edge relations between nodes i and j. Corresponding to α_{ij}^{lk}, β_{ij}^{lm} is the attention value of the edge information of nodes i and j. Finally, the ith node x representation at the lth layer can be computed by:

$$\mathbf{x}^{\mathbf{l+1}}_{\mathbf{i}} = \mathbf{h}^{\mathbf{l+1}}_{\mathbf{node_i}} \parallel \mathbf{h}^{\mathbf{l+1}}_{\mathbf{edge_i}} \tag{9}$$

$$\mathbf{h}^{\mathbf{l+1}}_{\mathbf{i}} = \text{relu}\left(\mathbf{W_{l+1}}\mathbf{x}^{\mathbf{l+1}}_{\mathbf{i}} + \mathbf{b_{l+1}}\right) \tag{10}$$

3.5. Output and learning

To better integrate contextual information for the input words, we apply a *Bi-LSTM* to obtain its contextual information ("Context Information Representation", as in Fig. 4) and accept its output hidden state h_i as the initial representation $\mathbf{h^0_i}$ of leaf node i. Then, another *Bi-LSTM* is applied to encode the aspect word with its intermediate hidden state as the initial representation h^0_a of the root. Finally, the knowledge graph tree is used to train *R-GAT*, its root representation h^l_a is passed through a fully connected softmax layer and mapped to probabilities of different sentiment polarities. Moreover, the Cross-Entropy with ℓ_2-regularization is the loss function to train the model.

4. Experiment and analysis

4.1. Datasets

We validate our model on four public sentiment analysis datasets (i.e., Twitter, Restaurant and Laptop, MAMS) that have been used in many previous studies [43,91,92], where each dataset is split by sampling 50% of each author's documents into the training set, and the remainder for testing. More statistical information about datasets is available in Table 2.

Twitter is an American micro-blogging and social networking service. It allows users to update messages of no more than 280 characters, also known as "tweets". With more than 175 million accounts worldwide, Twitter is one of the most popular and observed social media platforms [93], which enables users to read and post messages expressing their opinions about brands, celebrities, products, and public events [45,46,91,94]. **Restaurant** consists of coarse aspect categories and annotations of overall sentence polarity, allowing reviewers to rate restaurants for food taste, service, price, and atmosphere,etc [95,96]. **Laptop** contains over 3K English sentences that are tagged by experienced human annotators regarding aspect terms and their polarity [41,42,91,95]. **MAMS** is a popular and challenging dataset for aspect-based sentiment analysis task, where each sentence contains two aspects with different sentiment polarities [52,87,97].

Therefore, these datasets involve many fields such as social scenes, restaurants, services, environment, movies, electronic goods, etc. The scope of these datasets is relatively broad and cannot be summarized by a specific domain or a few domains of knowledge. Additionally, YAGO's knowledge base is largely based on Wikipedia, WordNet, and GeoNames, which cover a wide range of expertise. In this paper, we define the knowledge information captured by the YAGO knowledge base for sentiment analysis as domain knowledge to enhance the capture of sentiment polarity.

Table 2
Statistics of evaluation datasets.

Dataset	Positive		Neural		Negative	
	Train	Test	Train	Test	Train	Test
Twitter	1561	173	3127	346	1560	173
Restaurant	2164	728	637	196	807	196
Laptop	994	341	464	169	870	128
MAMS	3380	400	5042	607	2764	329

4.2. Experiment settings

To fairly compare the performance of our KDGN with the baseline models, the experimental hyperparameters in this paper are set as follows: Stanford CoreNLP parser [79] and Biaffine parser [80] are used as dependency parser to build the sentence tree, respectively. The word vector embedding is initialized with 300-dimensional BERT [90] vectors, and the dependency embedding is 300 and train all models using Adam [98] as the optimization function. To prevent overfitting, ℓ_2-regularization λ is applied between $\{0, 1{\times}10^{-5}, 1{\times}10^{-4}, 1{\times}10^{-3}, 1{\times}10^{-2}\}$. Learning rate is 1×10^{-3} and is stopped according to the performance of the validation set, and the dropout rate is 0.2. We conduct experiments with five different random seeds and report the average accuracy and Macro-F1. We implemented code using the PyTorch framework, and all models are trained with two GPUs (NVIDIA GeForce GTX 2080Ti).

4.3. Baseline models

To better validate the effectiveness of our KDGN, we select attention-based models and syntax-based models as the baseline models to make comparisons.

• **Attention-based models.** Target-Dependent LSTM, **TD-LSTM (2015)** utilize two target-dependent LSTM to integrate the connections between the target word and context words [54]. With the attentional mechanism, the Attention-based Long Short-Term Memory Network, **ATAE(2016)** reveals the sentiment polarity of a sentence determined by its content and highly related to the concerning aspect of that sentence [10]. [10]. Recurrent Attention on Memory, **RAM (2017)** captures sentiment features separated by a long distance by the multiple-attention mechanism [55]. Interactive Attention Networks, **IAN (2017)** learns attentions in the contexts and targets to generate the representations for targets and contexts separately [58]. Multi-Grained Attention Network, **MGAN (2018)** employs an aspect alignment loss to depict the aspect-level interactions among the aspects with the same context [59]. Aspect-Based Sentiment Analysis applies **BERT (2019)** [53] to capture the contextual word representations and additional generated text to perform the ABSA task.

• **Syntax-based models.** Syntax-based Attention Mechanism LSTM, **SynATT (2018)** introduces an attention mechanism incorporating syntactic information for ABSA task [56]. Phrase Recursive Neural Network, **PhraseRNN (2015)** extends the RNN by integrating both dependency and constituent trees of a sentence into account [99]. Graph Attention Networks, **GAT (2019)** is a graph neural network operating on graph-structured data with leveraging masked self-attention [92, 100]. Target-Dependent Graph Attention Network, **TD-GAT (2020)** explicitly mines the dependency relationship among words based on the dependency graph [8]. Aspect-Specific Graph Convolutional Networks, **AS-GCN (2019)** exploits syntactical information and word dependencies over the dependency tree [43]. Convolution Over a Dependency Tree, **CDT (2019)** builds a Bi-LSTM to learn representations of words in a sentence, and then enhances the embeddings with GCN based on the dependency tree [7]. Sentiment Dependencies Graph convolutional Networks, **SD-GCN (2020)** can effectively capture the sentiment dependency relationships between multi-aspects in the sentence with graph convolutional networks [101]. Relational Graph Attention Network, **R-GAT (2020)** is an optimized version of GAT, aggregating the information on the side [91]. Dual Graph Convolutional Networks, **DualGCN (2021)** fuses the complementarity of syntax structures and semantic correlations into a graph neural network simultaneously [102]. Knowledge enhanced Graph Convolutional Networks, **Sentic GCN (2022)** establishes graph neural networks by integrating SenticNet's sentiment knowledge to enhance the dependency graph of sentences [50]. **dotGCN (2022)** is an alternative structure to explicit dependency trees [97].

4.4. Results analysis

The overall performance of each model is shown in Table 3, from which we can draw the following conclusions:

(1) Compared to serialized models such as LSTM-based models, the structured models (e.g., LSTM+SynATT, DT-GAT, ASGCN, SD-GCN, GAT, and R-GAT) based on dependency trees perform better than the attention mechanism-based ones, which indicates that syntactic information plays a good role on the ABSA task.
(2) In the serialized models, capturing aspect–sentiment pairs using the attention mechanism (e.g., RAM, IAN, MGAN, and BERT) is more accurate and less time-consuming than capturing long-dependent information using RNN-based (e.g., ATAE-LSTM, DT-LSTM), and the parallelism is stronger.
(3) As far as structured models are concerned, they are mainly divided into Tree-LSTM-based tree neural networks (e.g., LSTM+SynATT, PhraseRNN) and graph neural networks with dependency information (e.g., DT-GAT, ASGCN, SD-GCN, GAT, and R-GAT) that exhibit the superiority of graph neural networks in capturing aspect–sentiment pairs.
(4) For models incorporating knowledge, for instance, DualGCN contains syntactic knowledge, and Sentic GCN integrates sentiment knowledge. Overall, Table 4 exhibits that the accuracy of the fused knowledge model is higher than that of structured-based modeling, which indicates that knowledge is valid and necessary for sentiment analysis. On all datasets, KDGN with fused domain knowledge outperforms R-GAT. It has been demonstrated that on ABSA task, domain knowledge (e.g., user habits, product's popularity, attribution) influences users' sentiment polarity towards the product or service [103]. Moreover, both structured models (e.g., SynATT, TD-GAT, SD-GCN) and fused knowledge models (e.g., DualGCN, Sentic GCN) apply syntactic trees (i.e. syntax tree and dependency tree) to model graph neural networks. Structure-based models are more conducive to capturing sentiment polarity than serialized models (e.g., ATAE, DT-LSTM, RAM, IAN, MGAN).
(5) Our KDGN model is superior to all other models except for Sentic GCN on Laptop and dotGCN on Twitter. Specifically, KDGN applies R-GAT with BERT on Knowledge-aware Dependency Graph integrating dependency labels information (which specifies collocation relationships between words), domain knowledge influenced information such as the popularity of goods, and syntax path. The experimental results demonstrate that domain knowledge, dependency labels information, and syntax path enhance the representation of the ABSA task. On one hand, dependency information can effectively capture sentiment pairs no matter how close or far apart the sentiment item and sentiment polarity is. On the other hand, the latent semantic information affecting the sentiment polarity is better extracted by combining dependency labels data, domain knowledge and yntax path, which has been demonstrated for other NLP tasks [104,105]. Furthermore, pre-trained language models such as BERT can also enhance the model's ability to capture latent knowledge.
(6) Table 3 demonstrates that different parser tools [106] (e.g., spaCy, CoreNLP, Biaffine, Stanza) perform differently on different datasets, which indicates that the different syntactic parser affects the robustness of models. dotGCN with spaCy works best on Twitter, KDGN with Biaffine outperforms on Restaurant and MAMS.

Table 3
Overall performance on four benchmark datasets.

Models	Twitter		Restaurant		Laptop		MAMA	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
ATAE (2015)	–	–	77.20	–	68.70	–	–	–
DT-LSTM (2016)	69.51	67.98	79.10	69.00	71.22	65.75	–	–
RAM (2017)	69.36	67.30	80.23	70.80	74.49	71.35	–	–
IAN (2017)	–	–	78.60	–	72.10	–		
MGAN (2018)	72.54	70.81	81.25	71.94	75.39	72.47	–	–
BERT (2019)	75.28	74.11	83.62	78.28	77.58	72.38	82.82	81.90
PhraseRNN (2015)	–	–	66.20	59.32	–	–	–	–
SynATT (2018)	–	–	80.45	71.26	77.57	69.13	–	–
GAT (2019)	71.67	70.13	78.21	67.17	73.04	68.11	–	–
ASGCN (2019)	72.15	70.40	80.77	72.02	75.55	71.05	–	–
TD-GAT (2020)	72.68	71.15	80.35	76.13	74.13	72.01	–	–
CDT (2019)	74.66	73.66	82.30	74.02	77.19	72.99	80.70	79.79
SD-GCN (2020)	–	–	83.57	76.47	81.35	78.34	–	–
R-GAT (2020)	76.15	74.88	86.60	81.35	78.21	74.07	81.75	80.87
DualGCN (2021)	76.04	74.91	86.77	81.62	80.63	77.36	–	–
Sentic GCN (2022)	76.22	74.90	86.94	81.62	**81.35**	77.90	–	–
dotGCN (2022)	**78.11**	**77.00**	86.15	80.37	81.03	**78.10**	85.95	84.44
Ours: KDGN	77.64	75.55	**87.01**	**81.94**	81.32	77.59	**86.13**	**84.56**

Table 4
Training KDGN without BERT, Bi-LSTM, dependency labels, KG, Syntax path, respectively.

Model	Twitter	Restaurant	Laptop
KDGN	**77.64**	**87.01**	**81.32**
w/o BERT	76.43	85.29	80.11
w/o Bi-LSTM	77.22	86.38	81.19
w/o Dependency labels	76.86	86.10	79.95
w/o KG	76.48	87.17	80.96
w/o Syntax Path	76.30	86.96	80.21

Table 5
Results on two dependency parsers: Stanford and Biaffine.

Model	Twitter	Restaurant	Laptop
CoreNLP	76.21	85.33	79.89
Biaffine	**77.64**	**87.01**	**81.32**

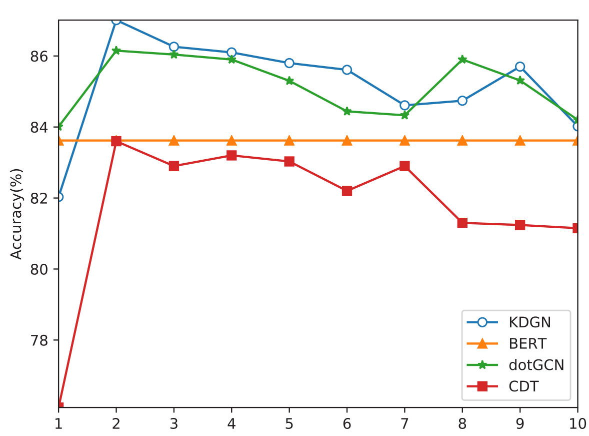

Fig. 5. Accuracy curves for BERT, CDT, dotGCN and KDGN on Restaurant.

4.5. Ablation studies

To better verify the impact of each component of KDGN, we also conduct ablation experiments. The results are shown in Table 4, in which "w/o" means "without". Table 4 shows that removing BERT, Bi-LSTM, dependency labels, KG, and syntax path all resulted in a significant decrease in the accuracy of the KDGN model. This indicates that all of the information fused in this paper can enhance the accuracy of the KDGN model on the ABSA task. Among them, removing the BERT has the most significant decrease in accuracy, followed by dependency labels, syntax path, KG and Bi-LSTM. From previous references [6,107], we can conclude that the Bi-LSTM model can effectively capture the contextual information of words, which further suggests that contextual semantic information impacts the expression of sentiment. KG is an enhanced representation of the explicit relational semantics of the sentiment pair. Furthermore, removing this information decreases the accuracy rate, indicating that domain information enhances sentiment analysis. In contrast, dependency labels explicitly represent the relationship between sentiment item and sentiment polarity, facilitating the explicit modeling of KDGN and simultaneously fusing this information into a high-dimensional vectorized representation. Finally, syntax path can better mine collocation information between words, especially those that apply multiple sentiment polarity words to modify sentiment items, which is very effective.

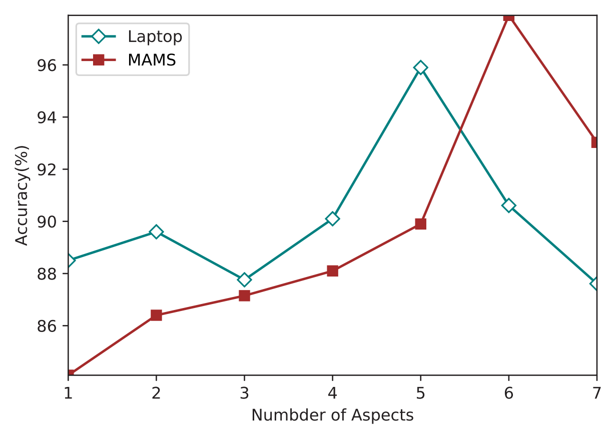

Fig. 6. Accuracy versus the number of aspects in the sentences.

4.6. Discussion

To further analyze which factors affect the performance of our KDGN model, we conduct three experiments, Effect of Different Parsers, Effect of GCN Layers and Effect of Multiple Aspects.

• **Effect of Different Parsers**. To verify the impact of dependent parsers on ABSA task, we evaluated our model with two well-known dependency parsers (i.e., Stanford CoreNLP Parser [108], Biaffine Parser) as examples. The experimental results in Table 5 manifest that different dependency parsers have an impact on ABSA task, which is consistent

#	Review	ATAE	Sentic GCN	dotGCN	KDGN
1	The **food** is very good, but the **location** is too far.	($P_{✓}$, $P_{×}$)	($P_{✓}$, $N_{✓}$)	($P_{✓}$, $N_{✓}$)	($P_{✓}$, $N_{✓}$)
2	I wish it had a **webcam** though , then it would be perfect.	$P_{×}$	$P_{×}$	$O_{×}$	$N_{✓}$
3	Apple's got a great **concept** but a little rough on the **delivery**.	($N_{×}$, $N_{✓}$)	(N×, $N_{✓}$)	($P_{✓}$, $N_{×}$)	($P_{✓}$, $N_{✓}$)
4	With the great variety on the **menu** , I eat here often, far from city, and **environment** never get bored.	($P_{✓}$, $N_{×}$)	($P_{✓}$, $N_{×}$)	($P_{✓}$, $P_{✓}$)	($P_{✓}$, $N_{✓}$)

Fig. 7. Case studies of our KDGN compared with state-of-the-art models (e.g., ATAE, Sentic GCN, dotGCN) for aspect-level sentiment classification, where ✓ and × refers to correct and incorrect predictions, respectively.

with existing studies [91,106]. Compared with CoreNLP, Biaffine shows better in capturing dependencies between sentiment items and sentiment polarity. However, when faced with complex textual information like sentiment analysis, especially some challenging datasets, an effective improvement of the dependency parser can improve the accuracy of the ABSA task.

• **Effect of GCN Layers**. To verify the effect of the number of GCN layers on the accuracy, we select BERT, CDT, dotGCN and KDGN for experiments on Restaurant dataset.

Fig. 5 illustrates that three convolutional neural networks (i.e., CDT, dotGCN, KDGN) combining dependency information in contextual cultural information achieve optimal performance over two layers. In comparison to serialized models (e.g., BERT), graph neural networks have a higher ability to capture information. Moreover, except for two layers, the accuracy of graph neural networks is not the best as the number of layers increases or decreases. This demonstrates that the number of GCN layers affects the accuracy of the ABSA task, which is consistent with the existing studies [7].

• **Effect of Multiple Aspects.** There are multiple aspect terms in a sentence on different datasets for ABSA task. Therefore, we intend to verify whether this phenomenon affects the effectiveness of our KDGN. We calculated the difference in training accuracy between the Laptop and MAMS datasets based on the number of aspect terms in each sentence. Since the sample size of items with more than 7 aspects is too small for meaningful comparison, samples with more than 7 aspects are excluded, which is consistent with existing work [36]. Fig. 6 shows that the accuracy of our KDGN fluctuates when the number of sentiment items in a sentence more than 3. This suggests that in future work, we need to build better models to capture the dependencies of multiple sentiment aspects in a sentence.

4.7. Case study

Fig. 7 shows several example cases analyzed using different models (e.g., ATAE, Sentic GCN, dotGCN), where the letters P, O and N represent positive, neutral and negative sentiments, respectively. We highlight the aspect words in red and blue for different sentiment polarities. For the first example sentence, the attentional mechanisms based on models (e.g., ATAE) are influenced by the proximity of "good". Although those constructed based on dependency relationships better capture the sentiment item and sentiment polarity pairs, they are still influenced by the semantic information of the whole sentence, as in example 2, which is a virtual hypothetical tone (e.g., "wish"), in fact the opposite of the true captured affective polarity. Sentic GCN integrates sentiment knowledge on the basis of dependency tree modeling to enhance the capture of sentiment pairs. Whereas dotGCN is modifying the distance between words without changing the relationship of words in the dependency tree, these models are more effective when there are multiple sentiment pairs. For example, the fourth example sentence is not optimized when there is one and only one sentiment pair. In contrast, our KDGN model can effectively capture the overall semantic information of the sentence by integrating BERT, Bi-LSTM, and syntax path, which can effectively capture the potential combination rules of words, especially sentiment terms.

Fig. 8. An example of attention weights obtained by KDGN integrating BERT, domain knowledge, dependency labels, and syntax path, respectively.

4.8. Attention visualization

To further verify the viability of the various components (e.g., BERT, dependency labels, domain knowledge, syntax path) of our KDGN model. We randomly select a sentence "Damn Chinese food is so good, especially Beijing food". from Restaurant and visualize it separately with the attention mechanism. The original sentence was changed by fusing BERT, dependency labels, and domain knowledge, as shown in Fig. 8. The five sentences in Fig. 8 are the original sentence, fused BERT, dependency labels, domain knowledge and syntax path, from top to bottom. The darker the color, the greater the attention value, indicating that the word is more relevant to the sentence.

Fig. 8 demonstrates that the critical words in sentence 1 are "food", "good" and "food", and integrating BERT, these critical words are "Chinese food", "so good" and "Beijing food", further demonstrating how BERT can improve sentence vectorized representation by incorporating contextual information of words. With the fusion of dependency labels, the vital information of sentence 3 is "Damn Chinese food", "so good","especially Beijing food" and the sentiment word "damn" increasing the polarity of the expression. For sentence 4 incorporating domain knowledge, the colors of "Chinese" and "Beijing" in "Damn Chinese food" and "particularly Beijing cuisine" are highlighted. This differs from the first three statements and indicates that domain knowledge plays a vital role in emphasizing and intensifying sentiment items. Example sentence 5 in Fig. 8 shows fused syntax path information that better captures collocation information between words, e.g., "Chinese"and "food", "Beijing"and "food". These attention distributions incorporating different information (i.e., dependency labels, domain knowledge, and syntax path) exhibit that the KDGN model proposed in this paper can effectively enhance the sentence representation, thus improving the accuracy of the ABSA task.

5. Conclusions and future work

This paper proposes a Knowledge-aware Dependency Graph Network that incorporates domain knowledge (e.g., popularity, fame), dependency labels and syntax path in the dependency tree. The experimental results show that our proposed KDGN model significantly improves the accuracy of the ABSA task on four benchmarking datasets (i.e., Twitter, Restaurant, Laptop, and MAMS). Moreover, we further validate the effect of domain knowledge, dependency labels and syntax path of the KDGN model in Ablation Studies and Case Study to demonstrate that these knowledge can enhance the accuracy of the model. In the future, for the issue of neutrality or ambivalence, we will incorporate more knowledge (e.g., semantic knowledge, syntax knowledge, inference knowledge) build complex inference models for the ABSA task, and also apply the neurosymbolic AI enhanced explainable sentiment analysis. Besides, we will verify the effect of syntactic parsing tools on model accuracy and robustness by constructing different syntactic models.

CRediT authorship contribution statement

Haiyan Wu: Design model, Collect datasets, Do experiments, Writing – original draft, Methodology, Writing – review & editing. **Chaogeng Huang:** Validation, Visualization, Investigation. **Shengchun Deng:** Conceptualization, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgments

The authors thank the three anonymous reviewers and the editors for their valuable comments and suggestions. This work is supported by the National Key Research and Development Program of China (No. 2019YFB1406300) and Zhejiang Provincial Natural Science Foundation of China under Grant (LY22F020027).

References

[1] C. Sun, L. Huang, X. Qiu, Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence, in: NAACL-HLT, Vol. 1, Association for Computational Linguistics, 2019, pp. 380–385.
[2] S. Kiritchenko, X. Zhu, C. Cherry, S. Mohammad, NRC-Canada-2014: Detecting aspects and sentiment in customer reviews, in: SemEval@COLING, The Association for Computer Linguistics, 2014, pp. 437–442.
[3] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, K. Xu, Adaptive recursive neural network for target-dependent Twitter sentiment classification, in: ACL, Vol. 2, The Association for Computer Linguistics, 2014, pp. 49–54.
[4] D. Tang, B. Qin, T. Liu, Document modeling with gated recurrent neural network for sentiment classification, in: EMNLP, The Association for Computational Linguistics, 2015, pp. 1422–1432.
[5] S. Liu, I. Lee, Sequence encoding incorporated CNN model for email document sentiment classification, Appl. Soft Comput. 102 (2021) 107104.
[6] H. Wu, Y. Liu, S. Shi, Modularized syntactic neural networks for sentence classification, in: EMNLP, Vol. 1, Association for Computational Linguistics, 2020, pp. 2786–2792.
[7] K. Sun, R. Zhang, S. Mensah, Y. Mao, X. Liu, Aspect-level sentiment analysis via convolution over dependency tree, in: EMNLP/IJCNLP, Vol. 1, Association for Computational Linguistics, 2019, pp. 5678–5687.
[8] L. Huang, X. Sun, S. Li, L. Zhang, H. Wang, Syntax-aware graph attention network for aspect-level sentiment classification, in: COLING, International Committee on Computational Linguistics, 2020, pp. 799–810.
[9] B. Huang, K.M. Carley, Syntax-aware aspect level sentiment classification with graph attention networks, in: EMNLP/IJCNLP, Vol. 1, Association for Computational Linguistics, 2019, pp. 5468–5476.
[10] Y. Wang, M. Huang, X. Zhu, L. Zhao, Attention-based LSTM for aspect-level sentiment classification, in: EMNLP, The Association for Computational Linguistics, 2016, pp. 606–615.
[11] B. Huang, K.M. Carley, Parameterized convolutional neural networks for aspect level sentiment classification, in: EMNLP, Association for Computational Linguistics, 2018, pp. 1091–1096.
[12] X. Hou, J. Huang, G. Wang, K. Huang, X. He, B. Zhou, Selective attention based graph convolutional networks for aspect-level sentiment classification, 2019, CoRR abs/1910.10857.
[13] R. Varghese, M. Jayasree, Aspect based sentiment analysis using support vector machine classifier, in: ICACCI, IEEE, 2013, pp. 1581–1586.
[14] M. Saeidi, G. Bouchard, M. Liakata, S. Riedel, SentiHood: Targeted aspect based sentiment analysis dataset for Urban neighbourhoods, in: COLING, ACL, 2016, pp. 1546–1556.
[15] I. Perikos, I. Hatzilygeroudis, Aspect based sentiment analysis in social media with classifier ensembles, in: 2017 IEEE/ACIS 16th International Conference on Computer and Information Science, ICIS, IEEE, 2017, pp. 273–278.
[16] M. Hu, B. Liu, Mining opinion features in customer reviews, in: AAAI, AAAI Press / The MIT Press, 2004, pp. 755–760.
[17] Y. Zhang, W. Zhu, Extracting implicit features in online customer reviews for opinion mining, in: WWW, Companion Volume, International World Wide Web Conferences Steering Committee / ACM, 2013, pp. 103–104.
[18] K. Schouten, F. Frasincar, Survey on aspect-level sentiment analysis, IEEE Trans. Knowl. Data Eng. 28 (3) (2016) 813–830.
[19] C. Fan, Q. Gao, J. Du, L. Gui, R. Xu, K. Wong, Convolution-based memory network for aspect-based sentiment analysis, in: SIGIR, ACM, 2018, pp. 1161–1164.
[20] Y. Ma, H. Peng, T. Khan, E. Cambria, A. Hussain, Sentic LSTM: A hybrid network for targeted aspect-based sentiment analysis, Cogn. Comput. 10 (4) (2018) 639–650.
[21] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, 2015, CoRR abs/1409.0473.
[22] M.S. Akhtar, D.K. Gupta, A. Ekbal, P. Bhattacharyya, Feature selection and ensemble construction: A two-step method for aspect based sentiment analysis, Knowl.-Based Syst. 125 (2017) 116–135.
[23] Q. Mao, J. Li, S. Wang, Y. Zhang, H. Peng, M. He, L. Wang, Aspect-based sentiment classification with attentive neural turing machines, in: IJCAI, 2019, pp. 5139–5145.
[24] W. Xue, T. Li, Aspect based sentiment analysis with gated convolutional networks, in: ACL, Vol. 1, Association for Computational Linguistics, 2018, pp. 2514–2523.
[25] A.P.B. Veyseh, N. Nouri, F. Dernoncourt, Q.H. Tran, D. Dou, T.H. Nguyen, Improving aspect-based sentiment analysis with gated graph convolutional networks and syntax-based regulation, in: EMNLP, Findings, in: Findings of ACL, vol.EMNLP 2020, Association for Computational Linguistics, 2020, pp. 4543–4548.
[26] K.S. Tai, R. Socher, C.D. Manning, Improved semantic representations from tree-structured long short-term memory networks, in: ACL, Vol. 1, The Association for Computer Linguistics, 2015, pp. 1556–1566.
[27] J. Li, T. Luong, D. Jurafsky, E.H. Hovy, When are tree structures necessary for deep learning of representations? in: EMNLP, The Association for Computational Linguistics, 2015, pp. 2304–2314.
[28] M. Song, H. Park, K. Shin, Attention-based long short-term memory network using sentiment lexicon embedding for aspect-level sentiment analysis in Korean, Inf. Process. Manag. 56 (3) (2019) 637–653.
[29] E. Jang, S. Gu, B. Poole, Categorical reparameterization With Gumbel-Softmax, Stat 1050 (2017) 5.
[30] J. Choi, K.M. Yoo, S. Lee, Learning to compose task-specific tree structures, in: AAAI, AAAI Press, 2018, pp. 5094–5101.
[31] Q. Li, D. Gkoumas, C. Lioma, M. Melucci, Quantum-inspired multimodal fusion for video sentiment analysis, Inf. Fusion 65 (2021) 58–71.
[32] K. Cho, B. van Merrienboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder-decoder approaches, in: SSST@EMNLP, Association for Computational Linguistics, 2014, pp. 103–111.
[33] A. Williams, S.R. Bowman, et al., Do latent tree learning models identify meaningful structure in sentences? Trans. Assoc. Comput. Linguist. 6 (2018) 253–267.
[34] Z. Lei, Y. Yang, M. Yang, W. Zhao, J. Guo, Y. Liu, A human-like semantic cognition network for aspect-level sentiment classification, in: AAAI, AAAI Press, 2019, pp. 6650–6657.
[35] A. Valdivia, M.V. Luzón, E. Cambria, F. Herrera, Consensus vote models for detecting and filtering neutrality in sentiment analysis, Inf. Fusion 44 (2018) 126–135.
[36] C. Zhang, Q. Li, D. Song, Aspect-based sentiment classification with aspect-specific graph convolutional networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP, 2019, pp. 4568–4578.

[37] H. Wu, Z. Zhang, S. Shi, Q. Wu, H. Song, Phrase dependency relational graph attention network for aspect-based sentiment analysis, Knowl.-Based Syst. 236 (2022) 107736.
[38] Y. Xiao, G. Zhou, Syntactic edge-enhanced graph convolutional networks for aspect-level sentiment classification with interactive attention, IEEE Access 8 (2020) 157068–157080.
[39] X. Chen, W. Zhang, X. Xu, W. Cao, A public and large-scale expert information fusion method and its application: Mining public opinion via sentiment analysis and measuring public dynamic reliability, Inf. Fusion 78 (2022) 71–85.
[40] A. Hussain, E. Cambria, S. Poria, A.Y.A. Hawalah, F. Herrera, Information fusion for affective computing and sentiment analysis, Inf. Fusion 71 (2021) 97–98.
[41] A.-S. Mohammad, M.M. Hammad, A. Sa'ad, A.-T. Saja, E. Cambria, Gated recurrent unit with multilingual universal sentence encoder for Arabic aspect-based sentiment analysis, Knowl.-Based Syst. (2021) 107540.
[42] C. Wu, Q. Xiong, Z. Yang, M. Gao, Q. Li, Y. Yu, K. Wang, Q. Zhu, Residual attention and other aspects module for aspect-based sentiment analysis, Neurocomputing 435 (2021) 42–52.
[43] C. Zhang, Q. Li, D. Song, Aspect-based sentiment classification with aspect-specific graph convolutional networks, in: EMNLP/IJCNLP, Vol. 1, Association for Computational Linguistics, 2019, pp. 4567–4577.
[44] A.B. Nassif, A. Elnagar, I. Shahin, S. Henno, Deep learning for arabic subjective sentiment analysis: Challenges and research opportunities, Appl. Soft Comput. 98 (2021) 106836.
[45] H. Tang, D. Ji, C. Li, Q. Zhou, Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification, in: ACL, Association for Computational Linguistics, 2020, pp. 6578–6588.
[46] Y. Tian, G. Chen, Y. Song, Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble, in: NAACL-HLT, Association for Computational Linguistics, 2021, pp. 2910–2922.
[47] H. Tang, D. Ji, C. Li, Q. Zhou, Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 6578–6588.
[48] A. Valdivia, M.V. Luzíón, F. Herrera, Neutrality in the sentiment analysis problem based on fuzzy majority, in: 2017 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE, IEEE, 2017, pp. 1–6.
[49] O. Araque, J.F. Sánchez-Rada, C.A. Iglesias, GSITK: A sentiment analysis framework for agile replication and development, SoftwareX 17 (2022) 100921.
[50] B. Liang, H. Su, L. Gui, E. Cambria, R. Xu, Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks, Knowl.-Based Syst. 235 (2022) 107643.
[51] A. Dai, X. Hu, J. Nie, J. Chen, Learning from word semantics to sentence syntax by graph convolutional networks for aspect-based sentiment analysis, Int. J. Data Sci. Anal. 14 (1) (2022) 17–26.
[52] X. Bai, P. Liu, Y. Zhang, Investigating typed syntactic dependencies for targeted sentiment classification using graph attention neural network, IEEE/ACM Trans. Audio Speech Lang. Proc. 29 (2021) 503.
[53] M. Hoang, O.A. Bihorac, J. Rouces, Aspect-based sentiment analysis using BERT, in: NODALIDA, Linköping University Electronic Press, 2019, pp. 187–196.
[54] D. Tang, B. Qin, X. Feng, T. Liu, Effective LSTMs for target-dependent sentiment classification, in: COLING, ACL, 2016, pp. 3298–3307.
[55] P. Chen, Z. Sun, L. Bing, W. Yang, Recurrent attention network on memory for aspect sentiment analysis, in: EMNLP, Association for Computational Linguistics, 2017, pp. 452–461.
[56] R. He, W.S. Lee, H.T. Ng, D. Dahlmeier, Effective attention modeling for aspect-level sentiment classification, in: COLING, Association for Computational Linguistics, 2018, pp. 1121–1131.
[57] D. Tang, B. Qin, T. Liu, Aspect level sentiment classification with deep memory network, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 214–224.
[58] D. Ma, S. Li, X. Zhang, H. Wang, Interactive attention networks for aspect-level sentiment classification, in: IJCAI, ijcai.org, 2017, pp. 4068–4074.
[59] F. Fan, Y. Feng, D. Zhao, Multi-grained attention network for aspect-level sentiment classification, in: EMNLP, Association for Computational Linguistics, 2018, pp. 3433–3442.
[60] F. Fan, Y. Feng, D. Zhao, Multi-grained attention network for aspect-level sentiment classification, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 3433–3442.
[61] H.K. Lin, T. Wang, G. Lin, S. Cheng, H. Chen, Y. Huang, Applying sentiment analysis to automatically classify consumer comments concerning marketing 4Cs aspects, Appl. Soft Comput. 97 (Part) (2020) 106755.
[62] X. Li, L. Bing, W. Zhang, W. Lam, Exploiting BERT for end-to-end aspect-based sentiment analysis, in: W-NUT@EMNLP, Association for Computational Linguistics, 2019, pp. 34–41.
[63] Y. Zhou, L. Liao, Y. Gao, R. Wang, H. Huang, TopicBERT: A topic-enhanced neural language model fine-tuned for sentiment classification, IEEE Trans. Neural Netw. Learn. Syst. (2021) 1–14.
[64] C. Zhang, Q. Li, D. Song, Syntax-aware aspect-level sentiment classification with proximity-weighted convolution network, in: SIGIR, ACM, 2019, pp. 1145–1148.
[65] X. Wang, G. Xu, J. Zhang, X. Sun, L. Wang, T. Huang, Syntax-directed hybrid attention network for aspect-level sentiment analysis, IEEE Access 7 (2019) 5014–5025.
[66] W. Ke, J. Gao, H. Shen, X. Cheng, Incorporating explicit syntactic dependency for aspect level sentiment classification, Neurocomputing 456 (2021) 394–406.
[67] W. Li, W. Shao, S. Ji, E. Cambria, BiERU: Bidirectional emotional recurrent unit for conversational sentiment analysis, Neurocomputing 467 (2022) 73–82.
[68] L. Zhu, X. Zhu, J. Guo, S. Dietze, Exploring rich structure information for aspect-based sentiment classification, J. Intell. Inf. Syst. (2022) 1–21.
[69] J. Zhou, J.X. Huang, Q.V. Hu, L. He, SK-GCN: Modeling syntax and knowledge via graph convolutional network for aspect-level sentiment classification, Knowl.-Based Syst. 205 (2020) 106292.
[70] A. Zhao, Y. Yu, Knowledge-enabled BERT for aspect-based sentiment analysis, Knowl.-Based Syst. 227 (2021) 107220.
[71] E. Cambria, Affective computing and sentiment analysis, IEEE Intell. Syst. 31 (2) (2016) 102–107.
[72] W. Zhang, J. Yan, Z. Wang, J. Wang, Neuro-symbolic interpretable collaborative filtering for attribute-based recommendation, in: Proceedings of the ACM Web Conference 2022, 2022, pp. 3229–3238.
[73] E. Cambria, Q. Liu, S. Decherchi, F. Xing, K. Kwok, SenticNet 7: A commonsense-based neurosymbolic AI framework for explainable sentiment analysis, Proc. LREC 2022 (2022) 3829–3839.
[74] J. Han, Z. Zhang, B.W. Schuller, Adversarial training in affective computing and sentiment analysis: Recent advances and perspectives [review article], IEEE Comput. Intell. Mag. 14 (2) (2019) 68–81.
[75] E. Cambria, S. Poria, A. Hussain, B. Liu, Computational intelligence for affective computing and sentiment analysis [guest editorial], IEEE Comput. Intell. Mag. 14 (2) (2019) 16–17.
[76] W. Liu, P. Zhou, Z. Zhao, Z. Wang, Q. Ju, H. Deng, P. Wang, K-BERT: enabling language representation with knowledge graph, in: AAAI, AAAI Press, 2020, pp. 2901–2908.
[77] W. Liu, P. Zhou, Z. Zhao, Z. Wang, Q. Ju, H. Deng, P. Wang, K-bert: Enabling language representation with knowledge graph, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, no. 03, 2020, pp. 2901–2908.
[78] M. de Marneffe, C.D. Manning, The stanford typed dependencies representation, in: CF+CDPE@COLING, Coling 2008 Organizing Committee, 2008, pp. 1–8.
[79] C.D. Manning, M. Surdeanu, J. Bauer, J.R. Finkel, S. Bethard, D. McClosky, The stanford CoreNLP natural language processing toolkit, in: ACL, System Demonstrations, The Association for Computer Linguistics, 2014, pp. 55–60.
[80] T. Dozat, C.D. Manning, Deep biaffine attention for neural dependency parsing, 2016, arXiv preprint arXiv:1611.01734.
[81] T. Rebele, F.M. Suchanek, J. Hoffart, J. Biega, E. Kuzey, G. Weikum, YAGO: A multilingual knowledge base from wikipedia, wordnet, and geonames, in: International Semantic Web Conference, Vol. 2, in: Lecture Notes in Computer Science, vol.9982, 2016, pp. 177–185.
[82] E. Agichtein, L. Gravano, Snowball: Extracting relations from large plain-text collections, in: Proceedings of the Fifth ACM Conference on Digital Libraries, 2000, pp. 85–94.
[83] R. Bunescu, M. Pasca, Using encyclopedic knowledge for named entity disambiguation, in: 11th Conference of the European Chapter of the Association for Computational Linguistics, 2006, pp. 9–16.
[84] E. Cambria, D. Olsher, D. Rajagopal, SenticNet 3: A common and common-sense knowledge base for cognition-driven sentiment analysis, in: AAAI, AAAI Press, 2014, pp. 1515–1521.
[85] H. Wu, Y. Liu, S. Shi, Modularized syntactic neural networks for sentence classification, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP, 2020, pp. 2786–2792.
[86] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (11) (1997) 2673–2681.
[87] A.P.B. Veyseh, N. Nouri, F. Dernoncourt, Q.H. Tran, D. Dou, T.H. Nguyen, Improving aspect-based sentiment analysis with gated graph convolutional networks and syntax-based regulation, in: Findings of the Association for Computational Linguistics: EMNLP 2020, 2020, pp. 4543–4548.
[88] R. Zhang, Z. Hu, H. Guo, Y. Mao, Syntax encoding with application in authorship attribution, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 2742–2753.
[89] Z. Wang, C. Zong, Phrase structure parsing with dependency structure, in: Proceedings of the 23rd International Conference on Computational Linguistics: Posters, Association for Computational Linguistics, 2010, pp. 1292–1300.
[90] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: NAACL-HLT, Vol. 1, Association for Computational Linguistics, 2019, pp. 4171–4186.
[91] K. Wang, W. Shen, Y. Yang, X. Quan, R. Wang, Relational graph attention network for aspect-based sentiment analysis, in: ACL, Association for Computational Linguistics, 2020, pp. 3229–3238.
[92] J. Chen, H. Hou, Y. Ji, J. Gao, T. Bai, Graph-based attention networks for aspect level sentiment analysis, in: ICTAI, IEEE, 2019, pp. 1188–1194.
[93] I. Anger, C. Kittl, Measuring influence on Twitter, in: Proceedings of the 11th International Conference on Knowledge Management and Knowledge Technologies, 2011, pp. 1–4.

[94] D. Cenni, P. Nesi, G. Pantaleo, I. Zaza, Twitter vigilance: A multi-user platform for cross-domain Twitter data analytics, NLP and sentiment analysis, in: SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI, IEEE, 2017, pp. 1–8.

[95] A.G. Pablos, M. Cuadros, G. Rigau, V3: unsupervised aspect based sentiment analysis for SemEval2015 task 12, in: SemEval@NAACL-HLT, The Association for Computer Linguistics, 2015, pp. 714–718.

[96] H. Wan, Y. Yang, J. Du, Y. Liu, K. Qi, J.Z. Pan, Target-aspect-sentiment joint detection for aspect-based sentiment analysis, in: AAAI, AAAI Press, 2020, pp. 9122–9129.

[97] C. Chen, Z. Teng, Z. Wang, Y. Zhang, Discrete opinion tree induction for aspect-based sentiment analysis, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2022, pp. 2051–2064.

[98] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: ICLR, Poster, 2015, pp. 11–12.

[99] T.H. Nguyen, K. Shirai, PhraseRNN: Phrase recursive neural network for aspect-based sentiment analysis, in: EMNLP, The Association for Computational Linguistics, 2015, pp. 2509–2514.

[100] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, Graph attention networks, Stat 1050 (2017) 20.

[101] P. Zhao, L. Hou, O. Wu, Modeling sentiment dependencies with graph convolutional networks for aspect-level sentiment classification, Knowl.-Based Syst. 193 (2020) 105443.

[102] R. Li, H. Chen, F. Feng, Z. Ma, X. Wang, E.H. Hovy, Dual graph convolutional networks for aspect-based sentiment analysis, in: ACL/IJCNLP, Vol. 1, Association for Computational Linguistics, 2021, pp. 6319–6329.

[103] P.K. Jain, R. Pamula, G. Srivastava, A systematic literature review on machine learning applications for consumer sentiment analysis using online reviews, Comp. Sci. Rev. 41 (2021) 100413.

[104] H. Wu, Z. Zhang, Q. Wu, Exploring syntactic and semantic features for authorship attribution, Appl. Soft Comput. 111 (2021) 107815.

[105] Z. Zhang, Y. Dong, H. Wu, H. Song, S. Deng, Y. Chen, Metapath and syntax-aware heterogeneous subgraph neural networks for spam review detection, Appl. Soft Compu. 128 (2022) 109438.

[106] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, C.D. Manning, Stanza: A python natural language processing toolkit for many human languages, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 2020, pp. 101–108.

[107] Y. Ma, H. Peng, E. Cambria, Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM, in: AAAI, AAAI Press, 2018, pp. 5876–5883.

[108] D. Chen, C.D. Manning, A fast and accurate dependency parser using neural networks, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, 2014, pp. 740–750.