Knowledge Graph Augmented Network Towards Multiview Representation Learning for Aspect-Based Sentiment Analysis

Qihuang Zhong, Member, IEEE, Liang Ding, Member, IEEE, Juhua Liu, Member, IEEE, Bo Du, Senior Member, IEEE, Hua Jin, and Dacheng Tao, Fellow, IEEE

Abstract—Aspect-based sentiment analysis (ABSA) is a fine-grained task of sentiment analysis. To better comprehend long complicated sentences and obtain accurate aspect-specific information, linguistic and commonsense knowledge are generally required in this task. However, most current methods employ complicated and inefficient approaches to incorporate external knowledge, e.g., directly searching the graph nodes. Additionally, the complementarity between external knowledge and linguistic information has not been thoroughly studied. To this end, we propose a knowledge graph augmented network (KGAN), which aims to effectively incorporate external knowledge with explicitly syntactic and contextual information. In particular, KGAN captures the sentiment feature representations from multiple different perspectives, i.e., context-, syntax- and knowledge-based. First, KGAN learns the contextual and syntactic representations in parallel to fully extract the semantic features. Then, KGAN integrates the knowledge graphs into the embedding space, based on which the aspect-specific knowledge representations are further obtained via an attention mechanism. Last, we propose a hierarchical fusion module to complement these multi-view representations in a local-to-global manner. Extensive experiments on five popular ABSA benchmarks demonstrate the effectiveness and robustness of our KGAN. Notably, with the help of the pretrained model of RoBERTa, KGAN achieves a new record of state-of-the-art performance among all datasets.

Index Terms—Knowledge graph, multiview learning, feature fusion, aspect-based sentiment analysis.

I. INTRODUCTION

A fine-grained task of sentiment analysis, aspect-based sentiment analysis (ABSA) has grown to be an active research task in the community of natural language understanding (NLU) [1], [2], [3]. In particular, ABSA refers to judging the sentiment polarities (e.g., positive, neutral, and negative) towards the given aspects, which are usually the target entities appearing in the sentence [4]. Taking the sentence “The food was good, but the service was poor.” as an example, as shown in Fig. 1(a), the goal of ABSA is to predict the polarities “positive” and “negative” for the aspects food and service, respectively.

Recent ABSA modeling approaches are mainly based on deep neural networks (DNNs) owing to the capability of automatically extracting semantic features [5]. Specifically, based on the type of learned feature representations, existing DNNs for ABSA can be classified into two groups: context-based methods [6], [7], [8] and syntax-based methods [9], [10], [11]. Context-based methods first employ convolutional neural networks (CNNs) or long short-term memory networks (LSTMs) to extract the features of aspects and context words and then use the attention mechanism to capture the aspect-specific contextual representations. In addition to context-based methods, syntax-based methods attempt to model the nonlocal dependency trees (a case in point is shown in Fig. 1(b)) of sentences with graph neural networks, e.g., graph convolutional networks (GCNs) to encode the syntactic information and syntactically connect the aspects with related opinion words [12].

More recently, given effective knowledge, e.g., linguistic and commonsense, for representation approaches in NLU tasks [13], [14], [15], researchers employ external knowledge to augment the semantic features in ABSA models [16], [17], [18], [19]. However, they make extensive modifications to model structures or objectives to encode the different kinds of knowledge, limiting the applicability of their methods to a broader range of tasks and knowledge types. For example, Zhou et al. [17] directly utilized the words (w.r.t. aspect terms in sentences) in knowledge graphs as the seed nodes and selected the related nodes to construct the subgraphs. While these subgraph-based methods [17], [20] have achieved remarkable performance, there are still some problems, e.g., the process of constructing subgraphs is usually relatively complex and would bring more computation, especially when

Manuscript received 12 January 2022; revised 7 September 2022; accepted 23 February 2023. Date of publication 28 February 2023; date of current version 15 September 2023. This work was supported in part by the National Natural Science Foundation of China under Grants 62076186 and 62225113, and in part by the Science and Technology Major Project of Hubei Province (Next-Generation AI Technologies) under Grant 2019AEA170. This work was done during Qihuang Zhong’s internship at JD Explore Academy. Our source code and final models are publicly available at https://github.com/WHU-ZQH/KGAN. Recommended for acceptance by Prof. J. Tang. (Corresponding Authors: Juhua Liu and Bo Du.)

Qihuang Zhong and Bo Du are with the National Engineering Research Center for Multimedia Software, Institute of Artificial Intelligence, School of Computer Science, Hubei Key Laboratory of Multimedia and Network Communication Engineering, Wuhan University, Wuhan, Hubei 430072, China (e-mail: zhongqihuang@whu.edu.cn; dubo@whu.edu.cn).

Liang Ding and Dacheng Tao are with the JD Explore Academy at JD.com, Beijing 100101, China (e-mail: liangding.liam@gmail.com; dacheng.tao@gmail.com).

Juhua Liu is with the Research Center for Graphic Communication, Printing and Packaging, Institute of Artificial Intelligence, Wuhan University, Wuhan, Hubei 430072, China (e-mail: liujuhua@whu.edu.cn).

Hua Jin is with the Hospital of Kunming University of Science and Technology, Kunming 650032, China (e-mail: jinhuakm@163.com).

Digital Object Identifier 10.1109/TKDE.2023.3250499
there are many aspect terms. Hence, we attempt to integrate external knowledge from a different perspective.

In this article, we propose a novel knowledge graph augmented network, namely, KGAN, to integrate external knowledge for boosting the performance of ABSA task. In general, KGAN employs three parallel branches to learn the feature representations from multiple perspectives (i.e., context-, syntax- and knowledge-based). The contextual and syntactic branches are used to extract the explicit context and syntax information from the labeled ABSA data, respectively, as most existing ABSA models do. More specifically, in the knowledge branch, unlike the above previous methods that usually employ complicated approaches to encode the knowledge, we recast them with a simpler and more efficient strategy to incorporate the external knowledge. In practice, instead of directly operating on graph-structure data, we first integrate external knowledge graphs into low-dimensional continuous embeddings, which can be simply and efficiently used to represent sentences and aspects. Then, based on the knowledge embeddings, a soft attention mechanism is utilized to capture the aspect-specific knowledge representations. As a result, we can obtain multiple representations that establish the relations between aspects and opinion words from different views. To take full advantage of the complementarity of these multiview representations, we introduce a novel hierarchical fusion module to effectively fuse them.

We conduct a comprehensive evaluation of KGAN on SemEval2014 (i.e., Laptop14 and Restaurant14), SemEval2015 (i.e., Restaurant15), SemEval2016 (i.e., Restaurant16) and Twitter benchmarks. The experimental results show that KGAN achieves comparable performance compared to the prior SOTA model with the GloVe-based setting. Moreover, we also investigate and demonstrate the effectiveness and robustness of our KGAN in BERT- and RoBERTa-based settings. In particular, based on RoBERTa, our model achieves the SOTA performance among all datasets in terms of accuracy and macro-F1 score. More specifically, compared to the prior SOTA models, the accuracy improvements of KGAN on Twitter, Restaurant15 and Restaurant15 datasets are up to 2.49%, 3.28% and 2.06%, respectively. Finally, we also compare KGAN with the other models in terms of latency and model size and prove that KGAN can achieve a good trade-off between efficiency and performance.

The main contributions can be summarized as follows:
1) We propose a novel knowledge graph augmented network (KGAN), where different types of information are encoded as multiview representations to augment the semantic features, thus boosting the performance of ABSA.
2) To achieve better complementarity between multiview features, we design a novel hierarchical fusion module to effectively fuse them.
3) Experiments on several commonly used ABSA benchmarks show the effectiveness and universality of our proposed KGAN. In combination with pretrained models, i.e., RoBERTa, we achieve new state-of-the-art performance on these benchmarks.

The rest of this article is organized as follows. In Section II, we briefly review the related works. In Section III, we introduce our proposed method in detail. Section IV reports and discusses our experimental results. Lastly, we conclude our study in Section V.

II. RELATED WORKS

A. Aspect-Based Sentiment Analysis

Benefiting from the representation learned from the training data, DNN-based ABSA models have shown promising performance compared to handcrafted feature-based models. We categorize them into two classes, e.g., context- and syntax-based methods.

First, considering the easily obtained contextual information, using CNNs [6], [21], [22], [23], [24] and LSTMs [7], [16], [25], [26], [27], [28] to extract the aspect-specific feature representations from context has become the mainstream approach for ABSA. In particular, owing to the ability to learn sequential patterns, the target-dependent LSTM (TD-LSTM) was proposed by Tang et al. [25] to capture the aspect information. TD-LSTM simplifies connecting the aspect with all context words, neglecting the effect of relative opinion words. Therefore, Wang et al. [26] improved upon the TD-LSTM by introducing an attention mechanism to explore the potential correlations between aspects and opinion words. In the study of Ma et al. [27], two separate LSTMs were used to encode the context and aspect terms, and then an interactive attention mechanism was further proposed to extract the more relevant information between the context and aspect features.

On the other hand, considering the complexity and inefficiency of LSTM-like sequential models, many studies have attempted to employ more efficient CNNs to capture the compositional structure and n-gram features. Xue and Li [21] proposed a gated convolution network to extract the contextual features and employed the gate mechanism to selectively output the final sentiment features. Huang and Carley [23] introduced two neural units, i.e., the parameterized filter and parameterized gate, to incorporate aspect information into CNN. Notably, in CNN-based methods, it is common to employ the average of aspect embeddings as the aspect representation, which would cause the loss of sequence information. To address this issue, Li

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
et al. [6] introduced a target-specific transformation component based on CNNs to better learn the target-specific representation. However, due to the challenge of multiple aspects with different polarities in a sentence, context-based models usually confuse the connections between aspects and related opinion words. To this end, most recent efforts focus on leveraging the syntactic structure of the sentence to effectively establish the connection [9], [10], [11], [12], [29], [30], [31]. In practice, syntactic dependency trees are introduced to represent the sentence, and then GNNs are used to model the dependency trees and encode the syntactic information. Zhang et al. [9] first utilized dependency trees to represent sentences and then proposed graph convolution networks (GCNs) to exploit syntactical information from dependency trees. Additionally, to better connect the aspect and opinion words syntactically, Wang et al. [12] presented a novel aspect-oriented dependency tree structure and employed a relational graph attention network to encode the tree structure. In addition, regarding sentences that have no remarkable syntactic structure, Pang et al. [30] introduced a multichannel GCN to optimally fuse syntactic and semantic information and their combinations simultaneously. Similarly, in the study of Li et al. [11], a dual GCN model that consists of SemGCN and SynGCN modules was used to take advantage of the complementarity of syntax structure and semantic correlations.

B. Incorporating External Knowledge

Since linguistic and commonsense knowledge can be beneficial to understanding natural language, incorporating this knowledge into deep learning models has become an active topic in many fields [13], [14], [32], [33], [34]. A case in point is the ERNIE [32], which employed the large-scale corpora and knowledge graphs to train a knowledge-enhanced pretraining language model. ERNIE experimentally achieves great performance on various knowledge-driven downstream tasks.

However, in the task of ABSA, the existing methods fall short in exploring the knowledge to augment the sentiment analysis. One main reason for this is that the above knowledge is not explicitly expressed in the ABSA datasets. Therefore, some recent studies attempt to incorporate external knowledge to alleviate this issue [16], [17], [18], [20], [35], [36], [37]. Wu et al. [35] proposed a unified model to integrate sentiment and structure knowledge with contextual representations for better performance. Zhou et al. [17] proposed jointly encoding syntactic information and external commonsense knowledge, where the knowledge was sampled via the individual nodes. Moreover, in the study of Xing et al. [34], a knowledge-enhanced BERT was introduced to obtain representations enhanced with sentiment domain knowledge to improve ABSA performance.

Following this line of research, we introduce knowledge graphs to explicitly provide external knowledge for ABSA. This idea is relatively similar to AR-BERT [20], which incorporates information on aspect-aspect relations in knowledge graphs to improve the performance of existing ABSA models. While AR-BERT [20] can achieve encouraging performance with the help of a large-scale knowledge graph, its main focus is on modeling aspect relations (captured by a complex method) from large knowledge graphs. In contrast, we start from the multiview learning perspective and propose a novel ABSA model that uses a simpler and more efficient strategy to model knowledge graphs. Additionally, instead of only integrating external knowledge with contextual or syntactic information, we synergistically combine the knowledge with both contextual and syntactic information to obtain richer feature representations and effectively boost the performance of sentiment analysis.

III. KNOWLEDGE GRAPH AUGMENTED NETWORK

A. Problem Formulation

In this section, we first define the task of ABSA mathematically. Suppose we have a sentence-aspect pair \( \{S, T\} \), where \( S = \{w_1, w_2, \ldots, w_m\} \) denotes the \( m \)-words sentence, and \( T = \{w_{\text{start}}, w_{\text{start}+1}, \ldots, w_{\text{start}+n-1}\} \) denotes the \( n \)-words aspect that is usually the subsequence of the sentence \( S \). Note that \( \text{start} \) is the starting index of \( T \) in \( S \). The goal of ABSA is to predict the sentiment polarity \( y \in \{0, 1, 2\} \) of the sentence \( S \) towards the aspect \( T \), where 0, 1, and 2 denote the positive, neutral and negative sentiment polarities, respectively.

B. Overview of the KGAN Model

The architecture of our proposed KGAN model is shown in Fig. 2, where KGAN contains three branches, i.e., context-, syntax- and knowledge-based branches, which learn the feature representations from multiple views. Specifically, the contextual and syntactic branches extract the contextual and syntactic features from the sentence represented by the pretrained word embeddings and explicitly establish the relevance between aspects and opinion words in the sentence. We then present the knowledge branch to model the introduced knowledge graphs and incorporate the external knowledge into the learned semantic features. In practice, knowledge graphs are first embedded into distributed representations, and a soft attention mechanism is then utilized to learn aspect-specific knowledge representations. Last, we synergistically fuse the learned multiview representations with a hierarchical fusion module.

C. Multi-View Representation Learning

1) Context-Based Representations: Recent works [38], [39] have shown that context-aware representation could successfully improve the language understanding ability, thus achieving better performance. Given the sentence-aspect pair \( \{S, T\} \), we employ the popular pretrained word embedding model to represent each word of \( S \) and \( T \), respectively. In particular, we embed each word \( w_i \) into a low-dimensional vector space with embedding matrix \( E \in \mathbb{R}^{V \times d_w} \), where \( V \) and \( d_w \) are the size of the vocabulary and the dimension of word embeddings, respectively. The embedding matrix is usually initialized with the embeddings of pretrained models, e.g., the static word embedding model GloVe [40]. Notably, inspired by the success of large-scale pretrained language models, e.g., BERT [41] and RoBERTa [42], we can also follow [12], [31] and use the BERT/RoBERTa as the word embedding extractor that can be also referred to as a more sophisticated “embedding matrix”. In practice, \( S \) and \( T \) are
Fig. 2. The architecture of our proposed knowledge graph augmented network (KGAN), which leverages external knowledge graphs to augment contextual and syntactic information. The \( R_c \), \( R_s \) and \( R_k \) denote the context- (left), syntax- (middle) and knowledge-based (right) representations, respectively. In the knowledge branch, ANALOGY and DistMult refer to the approaches of Knowledge Graph Embeddings (KGE). The GloVe/BERT/RoBERTa is used to convert the sentence/aspect into word embeddings. Based on the word embeddings, two separate bidirectional LSTMs (BiLSTMs) are used to capture the statistical dependencies in the sentence and aspect. In particular, we denote the forward operation of the LSTM as \( \rightarrow \text{LSTM} \) and the backward operation as \( \leftarrow \text{LSTM} \). The hidden state vectors \( h_s^i \) and \( h_t^j \) can be obtained with:

\[
\begin{align*}
  h_s^i &= \left[ \text{LSTM}(x_s^i), \text{LSTM}(x_s^i) \right], \quad i \in \{1, m\} \\
  h_t^j &= \left[ \text{LSTM}(x_t^j), \text{LSTM}(x_t^j) \right], \quad j \in \{1, n\}
\end{align*}
\]

As a result, we obtain the hidden output of BiLSTM for the sentence as \( H_s^o = \{h_1^s, h_2^s, \ldots, h_m^s\} \), which preserves the contextual information, and the target representation as \( H_c^i = \{h_1^c, h_2^c, \ldots, h_n^c\} \). In addition to \( H_s^o \), two attention mechanisms are introduced to capture the aspect-specific contextual features. Specifically, a self-attention mechanism is first used to fully learn the long-range dependencies of context. Then, we make the other soft attention mechanism assign the weight for each word of \( S \) towards the \( T \) and obtain the weighted aggregation as the aspect-specific contextual representation, namely, \( R_c \).

2) Syntax-Based Representations: In the syntactic branch, we aim to leverage the explicit syntactic information to encourage the model to learn the syntax-aware representations, denoted as \( R_s \), which has been shown to be helpful for many NLP tasks, e.g., machine translation [44], [45]. In practice, the same pretrained word embedding model and BiLSTM are also sequentially used to obtain the hidden state vectors \( H_s \). Note that we share the parameters of word embeddings and BiLSTM in both contextual and syntactic branches to reduce the computation and lighten the model size, i.e., \( H_s = H_s^c \).

Following the representative work in [9], we then employ a two-layer GCN module to extract the syntactic features of the sentence. To enable a close look at this, we show an illustration of the syntactic branch in Fig. 3.

First, we construct the syntactic dependency tree of the \( S \) with the spaCy toolkit1 and obtain the adjacency matrix, namely, \( A \), according to the words in the sentences. In practice, we make each word adjacent to its children’s nodes and itself and set the values of adjacency nodes to ones.

The GCN is further used to encode the syntactic information of \( G \) into \( H_s \). In our preliminary experiments, we found that a two-layer GCN performs better than a one-layer GCN, while

1https://spacy.io/
more layers did not improve the performance, which is consistent with results reported by previous works [46] and [47]. We claim that a one-layer GCN hardly captures the larger neighboring information, while multiple layers lead to high complexity. Therefore, the two-layer GCN is used last in our work. Specifically, with the operation of GCNs, the update of hidden state vectors $H_s$ can be formulated as follows:

$$
H_s^{(l+1)} = ReLU \left( \frac{A H_s^{(l)} W^{(l)} + B^{(l)}}{D + 1} \right), \quad l \in \{0, 1\}
$$

where $A$ is the adjacency matrix over the dependency tree, $D$ is the degree matrix of $A$ (i.e., $D_{ij} = \sum_j A_{ij}$), and $W^{(l)}$ and $B^{(l)}$ are the weight and bias matrices for the $(l+1)$-th GCN layer. $H_s^{(0)}$ is the initial hidden state vector $H_s$, and $H_s^{(2)}$ is the final output of the GCNs.

Additionally, we further introduce a graph-based attention module to learn the aspect-specific $R_s$. More specifically, the attention module first performs aspect masking on the top of $H_s^{(2)}$ to mask the nonaspect words with zero. Since the above GCNs perceived the important information in the hidden aspect state, masking the other states could alleviate the effect of noise. A dot-product attention mechanism is utilized to transition the related aspect-specific features from the initial $H_s$ towards the refined aspect features $H_s^{(2)}$ and thus syntactically build the connections of aspects and related opinion words.

3) Knowledge-Based Representations: To incorporate external knowledge and enrich the semantic features, we introduce the knowledge graphs of WordNet\(^2\) [48] as the external knowledge base, which contains more than 166,000 word form and sense pairs, and employs sets of synonyms to represent the concepts. There are several semantic relations between different concepts, e.g., similar, opposite, part, subordinate and entailment. As with human language acquisition, we first learn the basic (simple) concepts and progressively learn the abstract (difficult) concepts. For rare difficult words, we could comprehend their meanings via their relevant normal words. Inspired by this phenomenon, we employ WordNet as prior knowledge for sentence understanding. For example, the “Tinca” is the subordinate of the “fish genus,” which can be directly related to aspects such as “fish” or “food,” thus alleviating the difficulty of comprehending the sentence.

Different from Zhou et al. [17], which directly employs the graph-structure data of the knowledge base, we introduce a simple and efficient strategy to process the knowledge graphs. Specifically, semantic matching approaches (see the analysis of different approaches in Section IV-C3) for the task of knowledge graph embedding (KGE) [49] are used to model the semantic relations of knowledge graphs into distributed representations, i.e., learned knowledge embeddings. In practice, given the graph data in the form of “entity-relation-entity” triples, we train the entity embeddings using an open KGE toolkit, OpenKE\(^3\) [50]. Subsequently, we use the trained knowledge embeddings to initialize a new embedding matrix and then represent the words of $S$ and $T$ with the knowledge embedding matrix. The mapped knowledge embeddings are then concatenated with the hidden state vectors $H_s$. To establish the connection of $S$ and $T$ in knowledge embedding space, we further employ a soft attention mechanism to calculate the semantic relatedness of each word in $S$ and $T$ and capture the most important semantic features as aspect-specific knowledge representations, denoted as $H_k$. For better understanding, taking the sentence “Try the local food, especially the decent Tinca,” and the aspect word “food” as an example, the process of the knowledge branch is illustrated in Fig. 4. Notably, since the context word “Tinca” is the subordinate of the aspect “food” and they are also adjacent to each other in the knowledge embedding space, KGAN could easily capture their relatedness and make the correct prediction.

\(^2\)https://wordnet.princeton.edu/

\(^3\)https://github.com/thunlp/OpenKE.

\(^4\)Such a process can not only fuse the heterogeneous features (text and graph), but also alleviate the negative effect of sparsity and inaccuracy of knowledge embeddings.
TABLE I
STATISTICS OF EVALUATED ASPECT-LEVEL DATASETS

| Datasets     | Division | #Positive | #Negative | #Neutral |
|--------------|----------|-----------|-----------|----------|
| Laptop14     | Train    | 980       | 858       | 454      |
|              | Test     | 340       | 128       | 171      |
| Restaurant14 | Train    | 2159      | 800       | 632      |
|              | Test     | 730       | 195       | 196      |
| Twitter      | Train    | 1567      | 1563      | 3127     |
|              | Test     | 174       | 174       | 346      |
| Restaurant15 | Train    | 912       | 256       | 36       |
|              | Test     | 326       | 182       | 34       |
| Restaurant16 | Train    | 1240      | 439       | 69       |
|              | Test     | 469       | 117       | 30       |

Fig. 5. Illustration of the hierarchical fusion module.

Notably, it also should be noted that the training of these branches is not independent. Specifically, given an input sentence-aspect pair \( \{S, T\} \), the contextual and syntactic branches apply the same embedding matrix to convert the \( S \) and \( T \) into the corresponding word embeddings, while the knowledge branch uses another knowledge embedding matrix to map the input entities to knowledge embeddings. As a result, through the parallel processing of three branches, KGAN can capture the aspect-specific information from multiple perspectives simultaneously.

D. Hierarchical Fusion Module

Since the above representations \( \{R_c, R_s, R_k\} \) are obtained from different views, directly fusing them may scarcely take advantage of their complementarity. To this end, we adopt a hierarchical fusion module to synergistically fuse these representations in a local-to-global manner, which could effectively boost the performance. An illustration of this fusion module is shown in Fig. 5. For ease of illustration, we employ the “input” to represent the procedures of multiple branches.

In the local fusion procedure, we first concatenate two of the three feature representations in rows, i.e., \( [R_c; R_s], [R_c; R_k] \) and \( [R_s; R_k] \), where “;” denotes vector concatenation operator. The fused representations are fed into three separate fully connected layers to obtain the predicted sentiment features, denoted as \( R_{cs}, R_{ck} \) and \( R_{sk} \). It is noteworthy that we do not share the parameters of these fully connected layers. Subsequently, to make full use of the complementarity between multiple sentiment features, we further fuse them at the global level. Specifically, the obtained sentiment features are concatenated in columns, i.e., \( [R_{cs}, R_{ck}, R_{sk}]^T \), and we feed them into a 3*3 convolution layer to selectively incorporate these features.

Through the above local and global fusion procedures, we can make the feature representations benefit from each other step by step. In this way, external knowledge could be better integrated with contextual and syntactic information, thus achieving more promising performance.

Last, we cast the output of the convolution layer as the final sentiment prediction, namely, \( p \), and employ the following cross-entropy loss function to guide the optimization and training.

\[
L = -\sum_i \sum_j y_{ij} \log(p_{ij}),
\]

where \( i \) indexes the instance of the ABSA dataset, and \( j \) indexes the sentiment polarity.

IV. EXPERIMENTS

A. Datasets and Experimental Settings

Experiments are conducted on five public standard aspect-level datasets, i.e., Laptop14, Restaurant14, Twitter, Restaurant15 and Restaurant16. The Laptop14 and Restaurant14 datasets are from the SemEval2014 ABSA challenge [4], Restaurant15 and Restaurant16 are from SemEval2015 [51] and SemEval2016 [52] challenges respectively. The Twitter dataset is a collection of tweets [53]. Following [54], we remove a few instances with conflict sentiment polarity and list the final statistics of these datasets in Table I. Note that we evaluate the performance with respect to Accuracy (“Acc”) and Macro-F1 (“F1”).

In our implementation, we carefully validate the effectiveness of our KGAN based on three pretrained models, including GloVe\(^5\) [40] with 300 dimensions, BERT [41] and RoBERTa.\(^6\) We use the BERT-base-uncased and RoBERTa-base models with their default settings, i.e., 12 layers with 768-dimensional hidden vectors. The max sequence length for BERT/RoBERTa is 512. In particular, we employ them to convert the sentence and aspect into word embeddings. Following the existing work [55], we keep the knowledge embeddings fixed during training. The learning rates are empirically set as 1e-3 for GloVe-based KGAN, 5e-5 for KGAN-BERT and 3e-5 for KGAN-RoBERTa.

\(^5\)http://nlp.stanford.edu/data/glove.840B.300d.zip

\(^6\)We obtain the models from https://huggingface.co/models [42].
We set the batch size to 32 for most datasets, except 64 for Restaurant14. To avoid overfitting, we apply dropout on the word embeddings with a drop rate of 0.5. Adam [56] is employed to fulfill the optimization and training. The random seed is set to 14 in all experiments.

For comparison, we report other competitive approaches on different pretrained embeddings. Specifically, the GloVe-based methods can be roughly divided into three categories:

1) Context-based methods:
- ATAE-LSTM [26]: The aspect embedding and attention mechanism are utilized in the LSTM for aspect-level sentiment classification.
- RAM [58]: This method employs multiple attention and memory networks to capture the aspect-specific sentence representation.
- TNet-AS [6]: Using the CNN as the feature extractor, TNet-AS introduces a target-specific transformation component to better incorporate the target information into the representation.
- MGAN [65]: This network proposes to employ the coarse-grained aspect category classification task to enhance the fine-grained aspect term classification task and introduces a novel attention mechanism to align the features between different tasks.
- MCRR-SA [8]: Considering the importance of opinion spans, the network presents neat and effective multiple CRFs to extract the aspect-specific opinion spans.

2) Syntax-based methods:
- ASGCN & ASCNN [9]: ASGCN is the first work to represent sentences with dependency trees and adopt GCN to explore the syntactical information from the dependency trees for ABSA. ASCNN is a variant of ASGCN that replaces the 2-layer GCN with 2-layer CNN.
- R-GAT [12]: To better model syntax information, R-GAT proposes a novel aspect-oriented dependency tree structure to reshape and prune ordinary dependency parse trees.
- DGEDT [31]: The transformer is introduced into the network to diminish the error induced by incorrect dependency trees, thus boosting the performance.
- RGAT [60]: A relational graph attention network is proposed to make full use of the dependency label information, which is intuitively useful for the ABSA task.
- kumaGCN [24]: Regarding the problem of unsatisfactory dependency trees, kumaGCN associates dependency trees with induced latent aspect-specific graphs, and proposes gating mechanisms to dynamically combine information.
- DM-GCN [30]: Considering the lack of syntactic information in some bad cases, a dynamic and multichannel GCN is used to jointly model the syntactic and semantic structures for richer feature representations.
- DualGCN [11]: For the problem of inaccurate dependency parsing results, Dual-GCN leverages the additional semantic information to complement the syntactic structure.

3) External knowledge-based methods:
- Sentic-LSTM [61]: To explicitly leverage commonsense knowledge, this method proposes an extension of LSTM that can use the knowledge to control the information.
- MTKEN [35]: Multiple sources of knowledge, i.e., structure and sentiment knowledge, are fused in a unified model to boost the performance.
- SK-GCN [17]: A syntax- and knowledge-based GCN model is proposed to effectively incorporate syntactic information and commonsense knowledge by jointly modeling the dependency tree and knowledge graph.
- Sentic GCN [62]: This method constructs GCN via integrating the effective knowledge from SenticNet to enhance the dependency graphs of sentences.

Additionally, we compare KGAN to some powerful BERT- and RoBERTa-based methods to investigate the complementarity of our KGAN with powerful pretrained language models. Notably, despite our efforts to retrieve existing RoBERTa-based models, few models are applied to Restaurant15 and Restaurant16 datasets.

B. Main Results and Analysis

Table II lists the results of previous competitive models. First, in the GloVe settings, we find that our KGAN model outperforms the other cutting-edge methods on most evaluated datasets. Specifically, KGAN performs better than Sentic-LSTM and SK-GCN, which only combine external knowledge with single contextual or syntactic information, indicating the superiority of multiview representation learning. Additionally, in the Laptop14 dataset, compared to the current SOTA model DualGCN (Acc: 78.48%; F1: 74.74%), KGAN (Acc: 78.91%; F1: 75.21%) achieves performance improvements of 0.43% and 0.47% in terms of accuracy and macro-F1 score, respectively. Although the performance of KGAN (Acc: 84.46%; F1: 77.47%) for Restaurant14 is suboptimal, it also outperforms all models by at least 0.19% with respect to the accuracy metric and outperforms most models by over 1.39% in terms of macro-F1 score. These results can demonstrate the effectiveness and superiority of our KGAN.

Interestingly, we can find that the averaged performance of context-based models is worse than their syntax-based counterparts, especially on the Restaurant14 benchmark. One possible reason for this is that the ratio of multiaspect instances in Restaurant14 (26.58%) is higher than that of Laptop14 (20.05%) [28], where the nonlocal modeling ability provided by the syntactic dependent trees could effectively address such a multiaspect problem. More specifically, in the group of context-based methods, the CNN-based models, e.g., TNet-AS, significantly outperform the LSTM-based models, e.g., ATAE-LSTM and RAM, on the Twitter dataset. This is because instances of the Twitter dataset are almost ungrammatical and noisy [6], which greatly hinders the effectiveness of the LSTM.

---
7In preliminary experiments, we performed a grid search for the learning rate with {1e-3, 1e-4, 5e-5, 3e-5, 1e-5}, for the batch size with {16, 32, 64, 128}. The reported settings are selected for best practice.
Comparison With Previous Work on Different Datasets. Most results are retrieved from corresponding papers. "†", "‡" and "§" indicate the results are collected from [17], [12] and [57], respectively. "−" means the results are not reported in their original works. Notably, we report the averaged performance of KGAN with 5 random seeds to avoid stochasticity. The best results for each pretrained model are in bold, while the second-best results are underlined.

| Embedding Method | Laptop 14 | Restaurant 14 | Twitter | Restaurant 15 | Restaurant 16 |
|------------------|-----------|---------------|---------|---------------|---------------|
|                  | Acc. (%)  | F1 (%)        | Acc. (%)| F1 (%)        | Acc. (%)      |
| ATAB-LSTM [26]   | 68.88     | 63.99         | 78.60  | 67.02         | 68.64         |
| RAM [58]         | 74.49     | 71.35         | 80.23  | 70.80         | 69.36         |
| TNE+As [6]       | 76.54     | 71.75         | 80.69  | 71.27         | 74.97         |
| MGAN [59]        | 76.21     | 71.41         | 81.49  | 71.48         | 74.62         |
| MCRF SA [8]      | 77.64     | 74.23         | 82.86  | 73.78         | -             |
| GloVe            |           |               |        |               |               |
| ASCNN [9]        | 72.62     | 66.72         | 81.73  | 73.10         | 71.05         |
| ASCGCN [9]       | 75.55     | 71.05         | 80.86  | 72.19         | 72.15         |
| RGAT [12]        | 77.42     | 73.76         | 83.3   | 75.08         | 75.57         |
| DGEIDT [31]      | 76.80     | 72.30         | 83.90  | 75.10         | 74.80         |
| ROAT [60]        | 78.02     | 74.00         | 83.55  | 75.99         | 75.36         |
| kungaGCN [24]    | 76.12     | 72.42         | 81.43  | 73.64         | 72.14         |
| DM-GCN [30]      | 78.48     | 74.90         | 83.98  | 75.59         | 76.93         |
| DualGCN [11]     | 78.48     | 74.74         | 84.27  | 78.08         | 75.92         |
| Sentic-LSTM [61] | 70.88     | 67.19         | 79.43  | 70.32         | 70.66         |
| MTKRN [55]       | 73.43     | 69.12         | 79.47  | 69.08         | 69.54         |
| SK-GCN [17]      | 77.62     | 72.84         | 81.53  | 72.90         | 72.70         |
| Sentic GCN [62]  | 77.90     | 74.71         | 84.03  | 75.38         | -             |
| KGAN (Ours)      | 78.91     | 75.21         | 84.46  | 77.47         | 78.55         |
| BERT             |           |               |        |               |               |
| Vanilla BERT [41]| 77.58     | 72.38         | 85.62  | 78.28         | 75.28         |
| R-GAT-BERT [12]  | 78.21     | 74.07         | 86.60  | 81.35         | 76.15         |
| DGEIDT-BERT [31] | 79.80     | 75.60         | 86.30  | 80.00         | 77.90         |
| T-GCN-BERT [63]  | 80.88     | 77.03         | 86.16  | 79.95         | 76.45         |
| DM-GCN-BERT [30] | 80.22     | 77.28         | 87.86  | 82.79         | 77.59         |
| DualGCN-BERT [11]| 81.80     | 78.10         | 87.13  | 81.16         | 77.60         |
| Sentic GCN-BERT [62]| 82.12    | 79.05         | 86.92  | 81.03         | -             |
| KGAN-BERT (Ours) | 82.66     | 78.98         | 87.15  | 82.05         | 79.97         |
| RoBERTa          |           |               |        |               |               |
| Vanilla RoBERTa [43]| 83.78   | 80.73         | 87.37  | 80.96         | 77.17         |
| ASCGCN-RoBERTa [64]| 83.33  | 80.32         | 86.87  | 80.59         | 76.10         |
| RGAT-RoBERTa [12]| 83.33     | 79.95         | 87.52  | 81.29         | 78.15         |
| KGAN-RoBERTa (Ours)| 83.91   | 81.07         | 88.45  | 84.05         | 80.55         |

Last, we see that the performance of our KGAN on pretrained language models, i.e., BERT and RoBERTa, could achieve significant and consistent improvements compared with the results for GloVe, showing the complementarity between our approach and powerful pretrained language models. Encouragingly, our KGAN on RoBERTa achieves the new SOTA among all benchmarks, while the KGAN on BERT also outperforms most cutting-edge models in the same setting.

C. Ablation Study

In this section, we conduct extensive ablation studies to investigate the effects of multiple representations and the proposed fusion module in KGAN. Additionally, we analyze the influences of different knowledge graph embedding approaches. Unless otherwise stated, all mentioned KGAN models below are based on GloVe, and we believe that KGAN-BERT and KGAN-RoBERTa show similar trends.

1) Effects of Different Multiview Representation Combinations: Table III lists the results of different representation combinations, i.e., \{R_c, R_s, R_k\}. For fair comparison, we concatenate different representations and feed them into a one-layer MLP classifier to achieve multi-view fusion. Note that we do not employ the proposed hierarchical fusion module to fuse the entire representation combination \{R_c, R_s, R_k\}; thus, the performances of the last row in Table III are slightly worse than the relative ones in Table II.

2) Effects of Different Fusion Strategies: To validate the effectiveness of our proposed hierarchical fusion module (see

```markdown
| R_c  | R_s  | R_k  | Restaurant 14 | Twitter |
|------|------|------|---------------|---------|
| ✓    | ✓    | ✓    | 81.94         | 73.92   |
| ✓    | ✓    | ✓    | 81.42         | 72.85   |
| ✓    | ✓    | ✓    | 82.55         | 75.07   |
| ✓    | ✓    | ✓    | 82.81         | 75.00   |
| ✓    | ✓    | ✓    | 82.92         | 75.21   |
| ✓    | ✓    | ✓    | 83.35         | 75.25   |
```

Clearly, all representations from different views are of benefit to our KGAN. Specifically, R_k performs better than R_c and R_s, owing to the aligned features of text and graph data. As stated in Section III-C3, the knowledge embeddings (graph information) are concatenated with the hidden outputs of the BiLSTM (text information). Notably, with the help of R_k, the representation without knowledge \{R_c, R_s\} can achieve an averaged +1.08% improvement, showing the effectiveness of introducing knowledge to ABSA models. Recall that the combination of multiview representations \{R_c, R_s, R_k\} performs best among all combinations, and thus remains as the default setting.
fusing multiview representations (e.g., CONCAT, and SUM) is sub-optimal. In contrast, our proposed fusion module proposes to fuse these multiview representations in a local-to-global manner, which could take full advantage of their complementarity. Notably, it can be found that our method outperforms slightly than the VOTING (Acc: 78.13%; F1: 77.10%) on Twitter. One possible reason is that Twitter contains more rare and ungrammatical words, which are difficult to be completely covered by KGE, i.e., the knowledge embeddings for Twitter is relatively sparse and inaccurate, thus hindering the performance. Nevertheless, we state that when using a better KGE (e.g., that pretrained on larger knowledge graphs), the problem can be alleviated and the performance will be further improved (the detailed discussion can be found in Section IV-C3).

3) Effects of Different Knowledge Graph Embeddings: As mentioned above, we employ several simple Knowledge Graph Embedding (KGE) approaches to model the knowledge graphs into continuous embeddings. To further investigate how these knowledge embeddings affect the performance of KGAN, we conduct experiments to analyze the influences of a) different KGE approaches and b) different knowledge graphs.

For a), we use four typical approaches to embed the same knowledge graph (WordNet): a translated-based method TransE [66] and three semantic matching methods, i.e., ComplEx [67], ANALOGY [68] and DistMult [69]. Table IV (a) reports the experimental results. It can be found that the translated-based method TransE performs worse than the other semantic matching methods. Since TransE only focuses on encoding the relation information of entities, instead of semantic information, the knowledge embeddings learned by TransE fall short in enriching the semantic features of KGAN. Correspondingly, the models that could capture relational semantics are able to achieve better performance, especially ANALOGY (Acc: 78.91%; F1: 75.21%) on Laptop14. More interestingly, we find that the performance of DistMult is unstable. For Laptop14, the competitive DistMult method cannot achieve the optimal performance as it does for Restaurant14. One possible reason

![Figure 6](image_url)

**Fig. 6.** The performance in terms of accuracy (upper) and macro-F1 (bottom) for different fusion strategies. "CONCAT," "SUM," "ATT." and "VOTING" denote concatenation strategy, elementwise summation strategy, multi-view attention mechanism strategy and voting mechanism strategy, respectively.

As shown in Fig. 5, we compare it with several typical information fusion approaches: 1) "CONCAT": the multiview representations are directly concatenated in rows and fused by a fully connected layer; 2) "SUM": the representations are fed into three separate fully connected layers and fused via elementwise summation; 3) "ATT.": multi-view attention mechanism is used to fuse the multi-view representations. Specifically, given the representations \( \{R_c, R_s, R_k\} \), we first concatenate them to obtain the representation \( \left[ R_c, R_s, R_k \right]^T \) as the key and value in the attention mechanism. Sequentially, \( R_c/R_s/R_k \) are used as queries to calculate the attention scores respectively. These different view scores are added together and fed into a softmax layer to obtain the final attention scores, which are then used to multiply by the value. Lastly, the fused representation is fed into an MLP layer to obtain the sentiment prediction; 4) "VOTING": the representations are fused by voting mechanism, i.e., we train an individual classifier for each view representation and perform soft voting to determine the final prediction; 5) "OURS": the representations are fused by our proposed hierarchical fusion module.
for this is that DistMult falls short in modeling the semantic relation of the laptop domain. More potential reasons will be explored in future works.

To verify b), we introduce another larger knowledge graph, i.e., Wikidata, which contains more than 20 M entities and 594 relations. Due to the limited amount of computing resources, instead of training on Wikidata by ourselves, we use the publicly released Wikidata knowledge embeddings
\footnote{http://openke.thunlp.org/download/wikidata} pretrained with OpenKE toolkit. Notably, only two different dimension versions (50 and 100) of Wikidata knowledge embeddings trained with TransE are provided by OpenKE. For fair comparison, we use the same TransE to train the knowledge embeddings of WordNet and also set the embedding dimension as 50 and 100. The detailed results are listed in Table IV (b) as well.

When using the larger and higher-quality knowledge graphs, our KGAN consistently achieves better performance among all datasets. More specifically, the performance improvement on Twitter is up to 1.42% and 1.92% in terms of accuracy and macro-F1 score respectively. These results demonstrate that the larger and higher-quality knowledge graphs can further boost the performance of our KGAN. Moreover, it is noteworthy that the performance of Wikidata embeddings on Laptop14 and Restaurant14 is slightly worse than those WordNet embeddings trained with more sophisticated KGE approaches (e.g., ANALOGY and DistMult) in Table IV (a). We believe that our KGAN can be more advanced potentially, as the knowledge embeddings trained on large-scale knowledge graphs via more powerful KGE methods can bring more improvements\footnote{Since it is not our focus in this work and it is time-costly to pretrain the embeddings on such a large-scale knowledge graph, we do not conduct further experiments to empirically prove it.}.

### D. Case Study

For a closer look, we select cases from several evaluated datasets for extensive case studies. We first present some instances in Table V to show the effectiveness of our proposed KGAN. In practice, the aspects are enclosed in [ ], where the subscripts $p$, $n$, and $o$ denote the true polarities “positive,” “negative,” and “neutral,” respectively. Based on the table, we can obviously find that the context-based models (e.g., RAM and TNet-AS) perform worse than the syntax-based method (R-GAT), which shows that the syntax-based methods could effectively encode the syntactic information, thus better establishing the connections between aspects and related opinion words.

Moreover, our KGAN makes the correct predictions in most cases, indicating its effectiveness and superiority. In particular, the second sentence demonstrates that our KGAN model can effectively address complicated and informal sentences with the help of multiview representation learning. However, it should also be noted that all models predict erroneously in the last case, since the models are likely to focus on the opinion word “thick,” which misleads the comprehension of this sentence. Such a phenomenon indicates that KGAN requires further improvement to understand complicated sentences.

Additionally, we select two cases to show exactly how the multiview representations affect the KGAN. For both cases, we visualize the final attention weights of multiple branches and show the results in Figs. 7 and 8, respectively. The sentences are as follows:

1) The $[staff]_n$ should be a bit more friendly.

2) I never had an $[orange donut]_o$ before so I gave it a shot.

---

**TABLE V**

| Sentences                                                                 | RAM     | TNet-AS | MGAN   | R-GAT  | KGAN   |
|--------------------------------------------------------------------------|---------|---------|--------|--------|--------|
| Great $[food]_p$, but the $[service]_n$ is dreadful.                    | (P, N)  | (P, N)  | (P, N) | (P, N) | (P, N) |
| In mi burrito, here was nothing but dark $[chicken]_n$ that had          | (N, P:\) | (N, P:\) | (N, P:\) | (N, O:\) | (N, N) |
| that cooked last week and just warmed up in a microwave $[taste]_n$ .  |         |         |        |        |
| $[Startuptime]_n$ are incredibly long: over two minutes.                | $\n$    | N       | N      | N      | N      |
| The $[folding chair]_n$, I was seated at was uncomfortable.            | N       | O$\n$   | N      | N      | N      |
| The $[staff]_n$, should be a bit more friendly.                         | $\n$    | $\n$    | P$\n$  | P$\n$  | P$\n$  |
| It is really thick around the $[battery]_n$.                           | $\n$    | N$\n$   | N$\n$  | N$\n$  | N$\n$  |

---
In the first case, for the aspect “staff,” both representations from contextual and knowledge views focus on the opinion words “bit more friendly,” which facilitates the KGAN to understand the sentence and make the correct prediction. Moreover, we find that the representation from the syntactic view pays more attention to the other word “should,” which is also important for comprehension. These attention results demonstrate the complementarity of multiview representations.

On the other hand, in the second instance, the representation from the contextual view hardly captures the aspect-specific contextual information, while the representation from the syntactic view also fails to focus on the closely-related opinion words. However, the representation from the knowledge view can effectively extract the important related words “had” and “shot”. One possible reason for this is that the opinion words are not syntactically adjacent to the aspect, thus leading to the difficulty of capturing important semantic information. Instead, the “orange donut” is much closer to “had” in the introduced knowledge graph, which allows the knowledge branch to easily capture the relatedness. This case shows the significance of incorporating external knowledge and confirms our contribution.

E. Discussion

1) Comparison Between KGE-Based and Subgraph-Based Models: As stated in Section I, considering that current subgraph-based (or refer to GCN-based) methods usually require extra process to construct the aspect-specific subgraphs, which would lead to excessive computation, we attempt to integrating knowledge from another perspective, i.e., using KGE. Here, to verify whether KGE is more suitable for our KGAN, we compare the training efficiency and performance between KGE and current subgraph-based methods.

Specifically, we follow the currently existing works [17, 20] and provide three implementations to model the aspect subgraphs. For the first solution (denoted as K-KGAN), we simply replace the KGE module in the knowledge branch of KGAN with a two-layer GCN (similar to SK-GCN) [17], while keeping the other branches and feature fusion module unchanged. For the second method (denoted as SK-KGAN), we merge the syntactic dependency graph and aspect-specific knowledge graph and model them jointly with the GCN module in the syntax branch (similar to SK-GCN [17]), i.e., removing the knowledge branch. For the last one (denoted as AR-KGAN), inspired by AR-BERT [20], we use the popular GraphSAGE algorithm [70] to learn the aspect embeddings based on the subgraphs, which are used to replace the knowledge representation of knowledge branch in our KGAN.

Table VI shows the detailed results. Obviously, the KGE-based KGAN spends less time in data processing (smaller “DP. (s)” and requires less computation (smaller “GFLOPs”) compared to the other subgraph-based models, indicating the efficiency of KGE method. On the other hand, compared to the baseline “[R_c, R_s]” (without external knowledge), both KGE- and subgraph-based methods achieve better performance, continuing to confirm the significance of integrating external knowledge. To be more specific, it is noteworthy that our KGE-based KGAN achieves the best performance among all datasets, which shows its effectiveness. These results indicate that the KGE is indeed more suitable for our KGAN than current subgraph-based methods, but we also believe that it is good to keep in mind that subgraph-based methods can perform quite well in some scenarios when intrinsic relations in subgraphs can provide more valuable information both among implicit aspects and aspect-context correlations.

2) Latency and Model Size: Introducing external knowledge will admittedly increase the latency and model size [6, 12, 30]. We therefore perform a contrastive investigation on whether we achieve a good trade-off between efficiency and performance. For a fair comparison, all experiments are trained and tested on an Nvidia GTX-1660 SUPER. Table VII shows the details of KGAN and previous models, including ATAE-LSTM, R-GAT and DM-GCN. Their corresponding performances can be found in Table VI. Clearly, although our proposed KGAN model is more complicated than the previous context-based ATAE-LSTM, we can achieve significantly better performance (averaged +10.87% macro-F1 scores). Compared to more recent models, i.e., R-GAT and DM-GCN, our KGAN could achieve better performance while preserving comparable or better latency. The main reason for this is that these models introduced the sophisticated module to capture the semantic features, leading to larger model sizes, while we only employ the original BiLSTM as the feature extractor and share its parameters between three branches, greatly decreasing the model parameters. Takeaway: our KGAN establishes a good trade-off between efficiency and performance due to our parameter-sharing mechanism.

3) Robustness of Our KGAN: Some researchers may doubt that the incorporated external knowledge may function as the regularizer [71]. To dispel such doubt, we investigate whether

| Model     | Efficiency (avg.) | Latency | Model Size |
|-----------|-------------------|---------|------------|
|           | DP. (s) | PLOPs | Laptop14 | Restaurant14 |
| KGAN (Full) | 25.12 | 2.64 G | 78.91 | 75.21 | 84.46 | 77.47 |
| [R_c, R_s] | 22.72 | 2.60 G | 77.30 | 73.65 | 82.81 | 75.00 |
| K-KGAN    | 46.26 | 6.21 G | 77.97 | 73.99 | 82.12 | 73.76 |
| SK-KGAN   | 46.10 | 6.18 G | 77.66 | 73.92 | 82.90 | 74.42 |
| AR-KGAN   | 42.93 | 2.66 G | 77.64 | 74.16 | 83.25 | 75.77 |

**Table VI** COMPARISON OF KGE-BASED AND GCN-BASED KGAN ON THE TRAINING EFFICIENCY AND PERFORMANCE. “DP. (s)” DENOTES THE TIME OF DATA PROCESSING. “KGAN AND SK-KGAN ARE TWO GCN-BASED IMPLEMENTATIONS ON OUR KGAN.

| Model     | Latency (Train) | Latency (Test) | #Params (M) |
|-----------|-----------------|----------------|-------------|
| ATAE-LSTM | 1.98            | 0.71           | 2.53        |
| R-GAT     | 4.61            | 0.88           | 3.72        |
| DM-GCN    | 7.04            | 2.25           | 8.80        |
| KGAN (Ours) | 4.41          | 1.31           | 3.62        |

**Table VII** COMPARISON OF LATENCY AND MODEL SIZE WITH PREVIOUS WORKS ON RESTAURANT14. “#ARGS” IS THE NUMBER OF PARAMETERS IN MILLIONS. “TRAIN(s)” AND “TEST(s)” DENOTE THE AVERAGED TRAINING TIME (SECONDS) OF EACH EPOCH, AND THE INFERENCE TIME OF ALL TESTING SAMPLES, RESPECTIVELY.
our model is robust to noisy knowledge by introducing different percentages of noise to the knowledge embeddings on the Laptop14 dataset. Notably, noise attacks are widely used in the NLP community to investigate the robustness of models, such as neural machine translation [72, 73]. In particular, we randomly initialize some of the knowledge embeddings as noise. As shown in Fig. 9, our proposed multiview representation learning method can tolerate slight noise, e.g., 1%, 2% and 5%, and maintain performance to some extent. However, as noise increases, e.g., to 20%, the performance deteriorates, indicating that noisy knowledge does not function as regularization. Takeaway: 1) our multiview knowledge representation approach is robust to slight noise; 2) KGAN does not benefit from noise as much as it benefits from incorporated knowledge.

V. Conclusion

We present a novel knowledge graph augmented network for ABSA that incorporates external knowledge to augment semantic information. Specifically, KGAN captures the sentiment features from three different views: context, syntax and knowledge. These multiview feature representations are fused via a hierarchical fusion module. Extensive experiments demonstrate the effectiveness and robustness of our proposed KGAN. Ablation experiments and case studies show the complementarity between contextual, syntactic and external knowledge, confirming our claim. Extensive analyses illustrate that our KGAN can achieve a better trade-off between latency and performance and is robust to slight noise attacks.

Future work includes validating the proposed KGAN multiview representation approach in other challenging language understanding tasks, e.g., reading comprehension [74], intent identification and slot filling [75], and end-to-end language generation tasks [76], e.g., translation, summarization and grammatical error correction tasks.

REFERENCES

The numerical calculations in this article have been done on the supercomputing system in the Supercomputing Center of Wuhan University.

[1] B. Liu and L. Zhang, “A survey of opinion mining and sentiment analysis,” in Mining Text Data. Berlin, Germany: Springer, 2012, pp. 415–463.
[2] L. Zhang, S. Wang, and B. Liu, “Deep learning for sentiment analysis: A survey,” Wiley Interdiscip. Rev. Data Mining Knowl. Discov., vol. 8, no. 4, 2018, Art. no. e1253.
[3] K. Schouten and F. Frasincar, “Survey on aspect-level sentiment analysis,” IEEE Trans. Knowl. Data Eng., vol. 28, no. 3, pp. 813–830, Mar. 2016.
[4] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, “SemEval-2014 task 4: Aspect based sentiment analysis,” in Proc. 8th Int. Workshop Semantic Eval., Dublin, Ireland, 2014, pp. 27–35.
[5] K. Schouten and F. Frasincar, “Survey on aspect-level sentiment analysis,” IEEE Trans. Knowl. Data Eng., vol. 28, no. 3, pp. 813–830, Mar. 2016.
[6] X. Li, L. Bing, W. Lam, and B. Shi, “Transformation networks for target-oriented sentiment classification,” in Proc. 56th Ann. Meeting Assoc. Comput. Linguistics, Melbourne, Australia, 2018, pp. 946–956.
[7] B. Xu, X. Wang, B. Yang, and Z. Kang, “Target embedding and position attention with LSTM for aspect based sentiment analysis,” in Proc. 5th Int. Conf. Math. Artif. Intell., 2020, pp. 93–97.
[8] L. Xu, L. Bing, W. Lu, and F. Huang, “Aspect sentiment classification with aspect-specific opinion spans,” in Proc. ACM Conf. Empirical Methods Natural Lang. Process., 2020, pp. 3561–3567.
[9] C. Zhang, Q. Li, and D. Song, “Aspect-based sentiment classification with aspect-specific graph convolutional networks,” in Proc. ACM Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process., Hong Kong, China, 2019, pp. 4568–4578.
[10] X. Hou et al., “Graph ensemble learning over multiple dependency trees for aspect-level sentiment classification,” in Proc. ACM Conf. North Amer. Chapter Assoc. Comput. Linguistics Hum. Lang. Technol., 2021, pp. 2884–2894.
[11] R. Li, H. Chen, F. Feng, Z. Ma, X. Wang, and E. Hovy, “Dual graph convolutional networks for aspect-based sentiment analysis,” in Proc. ACM 59th Ann. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process., 2021, pp. 6319–6329.
[12] K. Wang, W. Shen, Y. Yang, X. Quan, and R. Wang, “Relational graph attention network for aspect-based sentiment analysis,” in Proc. ACM 58th Ann. Meeting Assoc. Comput. Linguistics, 2020, pp. 3229–3238.
[13] M. Yasunaga, H. Ren, A. Bosselut, P. Liang, and J. Leskovec, “QA-GNN: Reasoning with language models and knowledge graphs for question answering,” in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, 2021, pp. 535–546.
[14] Z. Liu, C. Xiong, M. Sun, and Z. Liu, “Entity-duet neural ranking: Understanding the role of knowledge graph semantics in neural information retrieval,” in Proc. 56th Ann. Meeting Assoc. Comput. Linguistics, 2018, pp. 2395–2405.
[15] S. Hu, L. Zou, J. X. Yu, H. Wang, and D. Zhao, “Answering natural language questions by subgraph matching over knowledge graphs,” IEEE Trans. Knowl. Data Eng., vol. 30, no. 5, pp. 824–837, May 2018.
[16] Y. Ma, H. Peng, and E. Cambria, “Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM,” in Proc. 32nd AAAI Conf. Artif. Intell., 2018, pp. 5876–5883.
[17] J. Zhou, J. X. Huang, Q. V. Hu, and L. He, “SK-GCN: Modeling syntax and knowledge via graph convolutional network for aspect-level sentiment classification,” Knowl.-Based Syst., vol. 205, 2020, Art. no. 106292.
[18] A. Zhao and Y. Yu, “Knowledge-enabled BERT for aspect-based sentiment analysis,” Knowl.-Based Syst., vol. 227, 2021, Art. no. 107220.
[19] K. Zhang et al., “EATN: An efficient adaptive transfer network for aspect-level sentiment analysis,” IEEE Trans. Knowl. Data Eng., vol. 35, no. 1, pp. 377–389, Jan. 2023.
[20] S. M. Islam and S. Bhattacharya, “AR-BERT: Aspect-relationship enhanced aspect-level sentiment classification with multi-modal explanations,” in Proc. ACM Web Conf., 2022, pp. 987–998.
[21] W. Xue and T. Li, “Aspect based sentiment analysis with gated convolutional networks,” in Proc. ACM 56th Ann. Meeting Assoc. Comput. Linguistics, Melbourne, Australia, 2018, pp. 2514–2523.
J. Basting, I. Titov, W. Aiziz, D. Marcheggiani, and K. Sima’an, “Graph convolutional encoders for syntax-aware neural machine translation,” in Proc. ACM Conf. Empirical Methods Natural Lang. Process., Copenhagen, Denmark, 2017, pp. 1957–1967.

Q. Wang, Z. Mao, and B. Wang, “Recurrent graph syntax encoder for neural machine translation,” 2019, arXiv:1908.06559.

T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in Proc. Int. Conf. Learn. Representations, 2016, pp. 1–14.

L. Yao, C. Mao, and Y. Luo, “Graph convolutional networks for text classification,” in Proc. AAAI Conf. Artif. Intell., 2019, pp. 7370–7377.

G. A. Miller, “WordNet: A lexical database for english,” Commun. ACM, vol. 38, no. 11, pp. 39–41, 1995.

P. Chen, Z. Sun, L. Bing, and W. Yang, “Recurrent attention network on memory for aspect-based sentiment analysis,” in Proc. Conf. Empirical Methods Natural Lang. Process. Syst. Demonstrations, 2018, pp. 139–144.

M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androusolopoulos, “SemEval-2015 task 12: Aspect based sentiment analysis,” in Proc. 9th Int. Workshop Semantic Eval., 2015, pp. 486–495.

M. Pontiki et al., “SemEval-2016 task 5: Aspect based sentiment analysis,” in Proc. Int. Workshop Semantic Eval., 2016, pp. 19–30.

L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, “Adaptive recursive neural network for target-dependent twitter sentiment classification,” in Proc. 52nd Annu. Meet. Assoc. Comput. Linguistics, 2014, pp. 49–54.

J. Tang et al., “Progressive self-supervised attention learning for aspect-level sentiment analysis,” in Proc. ACM 57th Annu. Meet. Assoc. Comput. Linguistics, Florence, Italy, 2019, pp. 557–566.

Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun, and Q. Liu, “ERNIE: Enhanced language representation with informative entities,” in Proc. 57th Annu. Meet. Assoc. Comput. Linguistics, Florence, Italy, 2019, pp. 1441–1451. [Online]. Available: https://aclanthology.org/P19-1139

W. Liu et al., “K-BERT: Enabling language representation with knowledge graph,” in Proc. AAAI Conf. Artif. Intell., 2020, pp. 2901–2908.

X. Yang, Z. Shi, Z. Meng, G. Lakemeyer, Y. Ma, and R. Wattenhofer, “KM-BART: Knowledge enhanced multimodal BART for visual commonsense generation,” in Proc. ACM 59th Annu. Meet. Assoc. Comput. Linguistics, 2021, pp. 525–535. [Online]. Available: https://aclanthology.org/2021.act-long-44

S. Wu, Y. Xu, F. Wu, Z. Yuan, Y. Huang, and X. Li, “Aspect-based sentiment analysis via fusing multi-sourced textual knowledge,” Knowl.-Based Syst., vol. 183, 2019, Art. no. 104868.

J. Liu, Q. Zhong, L. Ding, H. Jin, B. Du, and D. Tao, “Unified instance and knowledge alignment pretraining for aspect-based sentiment analysis,” 2021, arXiv:2110.13398.

Y. Wang, Q. Chen, M. Ahmed, Z. Li, W. Pan, and H. Liu, “Joint inference for aspect-level sentiment analysis by deep neural networks and linguistic hints,” IEEE Trans. Knowl. Data Eng., vol. 33, no. 5, pp. 2002–2014, May 2021.

L. Ding, L. Wang, D. Wu, D. Tao, and Z. Tu, “Context-aware cross-attention for non-autoregressive translation,” in Proc. 28th Int. Conf. Comput. Linguistics, 2020, pp. 4396–4402.

B. Yang, L. Wang, D. F. Wong, S. Shi, and Z. Tu, “Context-aware self-attention networks for natural language processing,” Neurocomputing, vol. 458, pp. 157–169, 2021.

J. Pennington, R. Socher, and C. Manning, “GloVe: Global vectors for word representation,” in Proc. ACM Conf. Empirical Methods Natural Lang. Process., Doha, Qatar, 2014, pp. 1532–1534.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, 2019, pp. 4171–4186.

Y. Liu et al., “RoBERTa: A robustly optimized bert pretraining approach,” 2019, arXiv:1907.11692.

D. Tang, B. Qin, and T. Liu, “Aspects level sentiment classification with deep memory network,” in Proc. ACM Conf. Empirical Methods Natural Lang. Process., Austin, Texas, 2016, pp. 214–224.
ZHONG et al.: KNOWLEDGE GRAPH AUGMENTED NETWORK TOWARDS MULTIVIEW REPRESENTATION LEARNING FOR ASPECT-BASED

[68] H. Liu, Y. Wu, and Y. Yang, “Analogical inference for multi-relational embeddings,” in Proc. Int. Conf. Mach. Learn., PMLR, 2017, pp. 2168–2178.

[69] B. Yang et al., “Embedding entities and relations for learning and inference in knowledge bases,” in Proc. Int. Conf. Learn. Representations, 2015, pp. 1–12.

[70] W. Hamilton, Z. Ying, and J. Leskovec, “Inductive representation learning on large graphs,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 1025–1035.

[71] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari, “Learning with noisy labels,” in Proc. Adv. Neural Inf. Process. Syst., 2013, pp. 1196–1204.

[72] S. Edunov, M. Ott, M. Auli, and D. Grangier, “Understanding back-translation at scale,” in Proc. Conf. Empirical Methods Natural Lang. Process., Brussels, Belgium, 2018, pp. 489–500. [Online]. Available: https://aclanthology.org/D18–1045

[73] L. Ding, L. Wang, and D. Tao, “Self-attention with cross-lingual position representation,” in Proc. ACM 58th Annu. Meeting Assoc. Comput. Linguistics, 2020, pp. 1679–1685. [Online]. Available: https://aclanthology.org/2020.acl-main.153

[74] Z. Zhang, Y. Wu, J. Zhou, S. Duan, H. Zhao, and R. Wang, “SG-Net: Syntax-guided machine reading comprehension,” in Proc. AAAI Conf. Artif. Intell., 2020, pp. 9563–9643.

[75] D. Wu, L. Ding, F. Lu, and J. Xie, “SlotRefine: A fast non-autoregressive model for joint intent detection and slot filling,” in Proc. Conf. Empirical Methods Natural Lang. Process., 2020, pp. 1932–1937.

[76] X. Liu, L. Wang, D. F. Wong, L. Ding, L. S. Chao, and Z. Tu, “Understanding and improving encoder layer fusion in sequence-to-sequence learning,” in Proc. Int. Conf. Learn. Representations, 2021, pp. 1–14.

Qihuang Zhong (Member, IEEE) is currently working toward the PhD degree in Artificial Intelligence with the School of Computer Science, Wuhan University. His research interests include language model pretraining, natural language understanding and generation. He has authored or coauthored several papers with top conferences and international journals, including IEEE Transactions on Knowledge and Data Engineering, EMNLP, COLING and etc. He won the general language understanding (GLUE) and more difficult language understanding (SuperGLUE) challenges.

Liang Ding (Member, IEEE) received the PhD degree from the University of Sydney. He is currently an algorithm scientist with JD.com and leading the NLP research group, JD Explore Academy. He works on deep learning for NLP, including language model pretraining, language understanding, generation, and translation. He published more than 30 research papers in NLP/AI, including ACL, EMNLP, NAACL, COLING, ICLR, AAAI, SIGIR, and CVPR, and importantly, some of his works were successfully applied to the industry. He served as area chair for ACL 2022 and session chair for SDM 2021 and AAAI 2023. He won many AI challenges, including SuperGLUE/GLUE, WMT2022, IWSLT 2021, WMT 2020, and WMT 2019.

Bo Du (Senior Member, IEEE) received the PhD degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China in 2010. He is currently a professor with the School of Computer Science and Institute of Artificial Intelligence, Wuhan University. He is also the director of the National Engineering Research Center for Multimedia Software, Wuhan University, Wuhan, China. He has more than 80 research papers published in the IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), IEEE Transactions on Image Processing (TIP), IEEE Transactions on Cybernetics (TCYB), IEEE Transactions on Geoscience and Remote Sensing (TGARS), etc. Fourteen of them are ESI hot papers or highly cited papers. His major research interests include machine learning, computer vision, and image processing. He serves as associate editor for Neural Networks, Pattern Recognition and Neurocomputing. He also serves as a reviewer of 20 Science Citation Index (SCI) magazines, including IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Cybernetics, IEEE Transactions on Geoscience and Remote Sensing, IEEE Transactions on Image Processing, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, and IEEE Geoscience and Remote Sensing Letters. He won the Highly Cited Researcher (2019/2020) by the Web of Science Group. He won IEEE Geoscience and Remote Sensing Society 2020 Transactions Prize Paper Award. He won the IJCAI (International Joint Conferences on Artificial Intelligence) Distinguished Paper Prize, IEEE Data Fusion Contest Champion, and IEEE Workshop on Hyperspectral Image and Signal Processing Best paper Award in 2018. He regularly serves as senior PC member of IJCAI and AAAI. He served as area chair for ICFPR.

Hua Jin received the MD and PhD degrees from Kunming Medical University. She is a professor of the affiliated hospital of Kunming University of Science and Technology, Kunming, China. She has authored and coauthored more than 50 scientific articles. Her past and present research interests include medical Big Data, computer vision and perioperative angesia.

Dacheng Tao (Fellow, IEEE) is currently the president of JD Explore Academy and a senior vice president of JD.com. He is also an advisor and chief scientist with the digital science institute, the University of Sydney. He mainly applies statistics and mathematics to artificial intelligence and data science. His research interests range across computer vision, data science, image processing, machine learning, and video surveillance. His research results have been presented in one monograph and 500+ publications in prestigious journals and at prominent conferences, such as IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on Image Processing, Journal of Machine Learning Research, IJCV, NIPS, ICML, CVPR, ICCV, ECCV, ICDM, and ACM SIGKDD, with several Best Paper awards, such as the Best Theory/Algorithm Paper Runner Up Award at IEEE ICDM07, the Best Student Paper Award at IEEE ICDM13, the Distinguished Student Paper Award at the 2017 IJCAI, the 2014 ICDM 10-Year Highest-Impact Paper Award, and the 2017 IEEE Signal Processing Society Best Paper Award. He was a recipient of the 2015 Australian Scopus-Eureka Prize, the 2015 ACS Gold Disruptor Award, and the 2015 UTS Vice-Chancellor’s Medal for Exceptional Research. He is a fellow of AAAS, OSA, IAPR, and SPIE.