

Exploiting Contextual Target Attributes for Target Sentiment Classification

Bowen Xing

BWXING714@GMAIL.COM

*Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing 100083, China
School of Computer and Communication Engineering,
University of Science and Technology Beijing, Beijing 100083, China*

Ivor W. Tsang

IVOR.TSANG@GMAIL.COM

*CFAR, Agency for Science, Technology and Research, 138632, Singapore
IHPC, Agency for Science, Technology and Research, 138632, Singapore
School of Computer Science and Engineering, Nanyang Technological University, 639798, Singapore
Australian Artificial Intelligence Institute, University of Technology Sydney
Ultimo, NSW 2007, Australia*

Abstract

In the past few years, pre-trained language models (PTLMs) have brought significant improvements to target sentiment classification (TSC). Existing PTLM-based models can be categorized into two groups: 1) fine-tuning-based models that adopt PTLM as the context encoder; 2) prompting-based models that transfer the classification task to the text/word generation task. Despite the improvements achieved by these models, we argue that they have their respective limitations. For fine-tuning-based models, they cannot make the best use of the PTLMs' strong language modeling ability because the pre-train task and downstream fine-tuning task are not consistent. For prompting-based models, although they can sufficiently leverage the language modeling ability, it is hard to explicitly model the target-context interactions, which are widely realized as a crucial point of this task. In this paper, we present a new perspective of leveraging PTLM for TSC: simultaneously leveraging the merits of both language modeling and explicit target-context interactions via *contextual target attributes*. Specifically, we design the domain- and target-constrained cloze test, which can leverage the PTLMs' strong language modeling ability to generate the given target's attributes pertaining to the review context. The attributes contain the background and property information of the target, which can help to enrich the semantics of the review context and the target. To exploit the attributes for tackling TSC, we first construct a heterogeneous information graph by treating the attributes as nodes and combining them with (1) the syntax graph automatically produced by the off-the-shelf dependency parser and (2) the semantics graph of the review context, which is derived from the self-attention mechanism. Then we propose a heterogeneous information gated graph convolutional network to model the interactions among the attribute information, the syntactic information, and the contextual information. The experimental results on three benchmark datasets demonstrate the superiority of our model, which achieves new state-of-the-art performance.

1. Introduction

The task of target sentiment classification (TSC) (Tang, Qin, Feng, & Liu, 2016; Wang, Pan, Dahlmeier, & Xiao, 2016a) is to predict the sentiment of a given target included in

a review. In the review “The noodles taste good, while the price is a little high.”, there are two targets (‘noodles’ and ‘price’) with positive and negative sentiments, respectively. Previous methods have realized that comprehensively understanding the review context and modeling the interactions between the context and the target are two crucial points for TSC. And much effort has been paid to enhance them, including leveraging syntactic information (Zhang, Li, & Song, 2019; Wang, Shen, Yang, Quan, & Wang, 2020; Li, Chen, Feng, Ma, Wang, & Hovy, 2021) and designing advanced attention mechanisms (Wang, Huang, Zhu, & Zhao, 2016b; Xing, Liao, Song, Wang, Zhang, Wang, & Huang, 2019; Chen, Sun, Bing, & Yang, 2017). Besides, in recent years, since pre-trained language models (PTLMs) like BERT (Devlin, Chang, Lee, & Toutanova, 2019) have brought stunning improvement in a bunch of NLP tasks, current TSC models widely adopt the pre-trained BERT to encode the review context and then obtain the high-quality context hidden states, which have been proven to boost the performance significantly. Most existing state-of-the-art (SOTA) TSC models are built on the BERT+Syntax+Attention framework. More recently, prompt learning has attracted increasing attention in NLP fields. It can transfer the classification task to the masked language modeling task. In this way, the best potential of PTLMs can be utilized because the pre-training task and the downstream task are consistent (both are masked language modeling tasks) (Liu, Yuan, Fu, Jiang, Hayashi, & Neubig, 2023). And prompting-based methods have been explored in TSC (Li, Lin, Chen, Dong, & Liu, 2022; Seoh, Birle, Tak, Chang, Pinette, & Hough, 2021).

Therefore, existing PTLM-based models can be categorized into two groups: 1) fine-tuning-based models that adopt PTLM as the context encoder; 2) prompting-based models that transfer the classification task to the text/word generation task. Despite the promising progress achieved in recent years, we argue that the two groups of models have their respective limitations. For fine-tuning-based models, it cannot make the best use of the PTLMs’ strong language modeling ability because the pre-train task and downstream fine-tuning task are not consistent. Although prompting-based models can sufficiently leverage the language modeling ability, it is hard to explicitly model the target-context interactions, which are widely realized as the key point of this task.

In this paper, we simultaneously leverage the merits of masked language modeling (MLM) and explicit target-context interactions. Unlike prompting-based methods leveraging MLM to generate the masked label word in the designed template, we mask the target in the review and obtain the generated words. And in this paper, we refer to this process as the target cloze test. After pre-training, the PTLMs are quite good language models, which can precisely generate the masked word fitting the semantics of the context well. Given this fact, we believe the generated words contain the target’s attribute information, which can help comprehensively understand the review context.

To investigate this, we conduct the target cloze test on some samples, and we list two of them in Table 1. We can find that in the first sample, the generated words are generally related to the target *Windows 7* and convey its property and background information. However, in the second sample, the generated words seem random and irrelevant to the target *browser*. We attribute this to the fact that in the first sample, there is semantic constrain from *Macbook pro laptop* and *vmware program*, while there is none in the second sample. Inspired by this discovery, we design the *domain- and target-constrained cloze test*, which aims to generate the target-related candidate words that potentially provide the

<p>1. Review: I also enjoy the fact that my Macbook pro laptop allows me to run [Windows 7]^{masked} on it by using the vmware program. Generated Words: games, linux, software, programs, applications, directly, ...</p>
<p>2. Review: Speaking of the [browser]^{masked}, it too has problems. Generated Words: past, sea, moon, weather, future, dead ...</p>

Table 1: Two examples of the target cloze test, which is the vanilla cloze test masking the target. And the frozen BERT_{base} is used to generate the candidate words.

target’s property and background information. In this paper, we refer to these generated words as *contextual target attributes* since they are generated regarding the review context and contain the target’s attribute information. We believe these contextual target attributes can benefit TSC via enhancing the context and target understanding.

To exploit contextual target attributes for TSC, we propose a new model whose core is to model the interactions of three kinds of information: (1) the target’s property and background information conveyed in the contextual target attributes; (2) the review’s syntactic information conveyed by the syntax graph; (3) the review’s contextual word correlative information conveyed by the fully-connected semantic graph derived from the self-attention mechanism. Specifically, we first propose a heterogeneous information graph (HIG), which includes two kinds of nodes: attributes nodes and context nodes. The connections among the context nodes are based on both the syntax graph and the semantic graph. Moreover, the attribute nodes not only connect to the target node(s) but also connect to some context words via syntax-heuristic edges. HIG provides a platform for attribute information-enhanced target-context interactions. On HIG the fine-trained and heterogeneous interactions among the target’s attribute information, target semantics and context semantics can be modeled. To model the heterogeneous information interactions on HIG, inspired from (Xing & Tsang, 2022b, 2022a, 2022d, 2023c, 2023a, 2023b), we propose a novel heterogeneous information gated graph convolutional network (HIG²CN). It includes a target- and context-centric gate mechanism to control the information flow. Besides, relative position weights are adopted to highlight the potential target-related words. In addition, we propose a heterogeneous convolution layer for information aggregation. Generally, HIG²CN has three advantages:

- It leverages the contextual target attributes to enhance the context and target understanding, capturing more beneficial clues via the heterogeneous information interactions.
- It integrates the structures of both syntax graph and semantic graph. Thus, it can capture more comprehensive and robust context knowledge.
- Due to its gate mechanism, more and more crucial information can be discovered and retained in the nodes and then aggregated into the target node(s).

In this way, our method can simultaneously leverage the MLM via the target cloze test and explicitly model the target-context interactions via HIG²CN. We conduct evaluation experiments on three benchmark datasets. The results show that our model outperforms existing models by a large margin, achieving new state-of-the-art performance.

2. Related Works

Different attention mechanisms (Wang et al., 2016b; Ma, Li, Zhang, & Wang, 2017; Chen et al., 2017; Tang, Lu, Su, Ge, Song, Sun, & Luo, 2019; Xing et al., 2019) have been designed to capture the target-related semantics. For example, IAN (Ma et al., 2017) utilizes the interactive attention mechanism to model the bidirectional interactions between the target and context. RAM (Chen et al., 2017) adopts a recurrent attention mechanism that calculates the attention weights recurrently to generally aggregate the target-related semantics.

Although attention mechanisms have achieved great improvements in TSC, in many cases, they still cannot capture the target-related semantics contained in some context words. Recently, researchers discovered that the distance between the target and its related words can be shortened by applying graph neural networks (GNNs) to encode the syntax graph of the review context, which is a promising way to facilitate capturing the beneficial clues. Therefore, different GNN-based models (Zhang et al., 2019; Tang, Ji, Li, & Zhou, 2020; Huang, Sun, Li, Zhang, & Wang, 2020; Tian, Chen, & Song, 2021a; Li et al., 2021; Xing & Tsang, 2022e; Wang et al., 2020; Xing & Tsang, 2022f, 2022c) are proposed to model the syntax graph capturing the useful syntactic information that conveys the dependencies between the target and the crucial words. For instance, ASGCN (Zhang et al., 2019) applies a multi-layer graph convolutional network (GCN) on the syntax graph, which is obtained from an off-the-shelf dependency parser, to extract the syntactic information. DualGCN (Li et al., 2021) is based on a dual-GCN architecture that captures both the semantic and syntactic information comprehensively. DotGCN (Chen, Teng, Wang, & Zhang, 2022) integrates the attention scores and syntactic distance into a discrete latent opinion tree.

However, previous works leverage PTLM only for encoding. Our work makes the first attempt to exploit the contextual target attributes from PTLM. Besides, we propose the HIG and HIG²CN to model the sufficient interactions among (1) the target’s background and property information contained in the attributes, (2) the semantics information and (3) syntactic information.

3. Overview

The overall framework of our method is illustrated in Fig. 1. In the preprocessing process, our proposed domain- and target-constrained cloze test is achieved by frozen BERT_{base} and conducted on all samples to obtain the contextual target attributes, which are used to construct HIG. Like previous methods, we utilize a off-the-shelf dependency parser¹ to produce the syntax graphs of the samples. In the training/testing process, after the BERT Encoder, the self-attention produces the attention matrix, which is adopted as the adjacent matrix of the fully-connected semantic graph. Then the attributes, syntax graph, and semantic graph are fed to the HIG Construction module to generate the HIG, on which HIG²CN is applied for heterogeneous information interactions.

Next, before diving into our model, we first introduce our proposed domain- and target-constrained cloze test.

1. <https://spacy.io/>

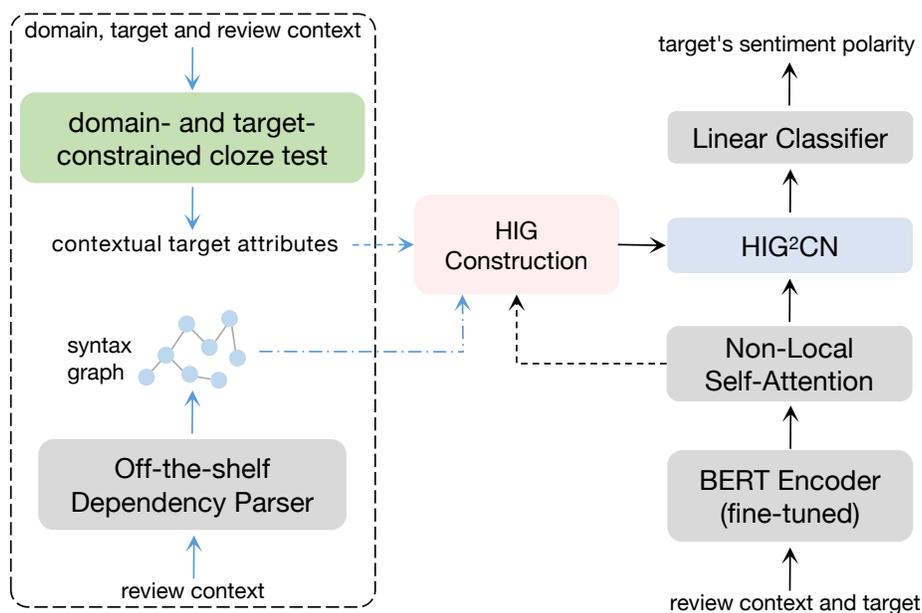


Figure 1: Overall architecture of our model. The dashed box denotes that the operations are conducted in the preprocessing procedure. The black dash arrow denotes the fully-connected semantic graph derived from the self-attention mechanism. HIG denotes heterogeneous information graph, and HIG²CN denotes heterogeneous information gated graph convolutional network.

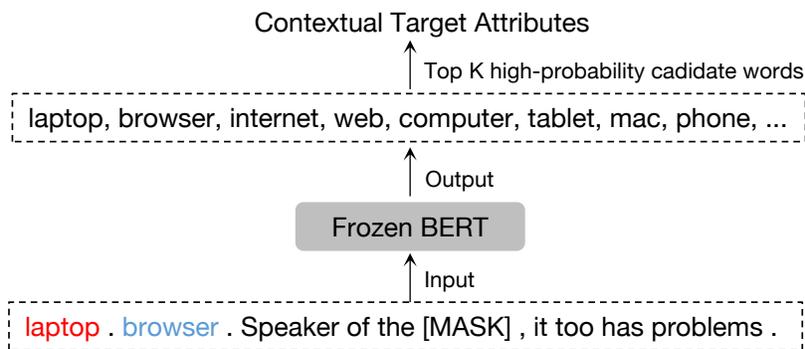


Figure 2: An example illustrating how to obtain the contextual target attributes. The red color denotes the domain word, and the blue color denotes the target. K denotes the number of obtained contextual target attributes.

4. Domain- and Target-Constrained Cloze Test

In this paper, we propose the domain- and target-constrained cloze test to obtain the contextual target attributes. An example illustrating this process is shown in Fig. 2. To apply the semantics constraint from the domain and target to the input sentence, we design a simple while effective template in which the domain word and target are concatenated to the beginning of the masked review context. From the second sample in Table 1, we can observe

that the general cloze test conducted on BERT generates random and irrelevant words that cannot be used as attributes. Differently, in Fig. 2, we can find that the domain- and target-constrained cloze test can generate the target-related words (e.g., laptop, internet, web), which we term as *contextual target attributes* conveying the background and property information of the target. Besides, they have another advantage that they naturally well-fit the context semantics since they are generated from the review context by the powerful PTLM. Therefore, they have the potential to enhance target and context understanding. Besides, the domain- and target-constrained cloze test is conducted on the frozen BERT, so this process can be conducted in the preprocessing process without costing extra training time.

5. Our Model

5.1 Encoding

BERT (Devlin et al., 2019) is a popular pre-trained language model based on the multi-layer Transformer architecture. In this paper, BERT is employed for context encoding to obtain the context hidden states. Denote the review context as $\{x_1, x_2, \dots, x_{N_c}\}$ and the target words as $\{t_1, \dots, t_{N_t}\}$, the input of BERT is the concatenation of these two sequences:

$$\langle [\text{CLS}]; x_1, x_2, \dots, x_{N_c}; [\text{SEP}]; t_1, \dots, t_{N_t}; [\text{SEP}] \rangle \quad (1)$$

where $[\text{CLS}]^2$ and $[\text{SEP}]$ are the special tokens in BERT; $\langle ; \rangle$ denotes sequence concatenation; N_t and N_c denote the word number of the target and review context, respectively. We take the last-layer representation in BERT as the context word hidden states: $\hat{H}_c = [\hat{h}_1^c, \dots, \hat{h}_{N_c}^c] \in \mathbb{R}^{N_c \times d}$, which includes the target hidden states $\hat{H}_t = [\hat{h}_1^t, \dots, \hat{h}_{N_t}^t] \in \mathbb{R}^{N_t \times d}$, where d denotes the hidden state dimension. Besides, due to BERT’s sentence-pair modeling ability, it has been proven that the $[\text{CLS}]$ token’s output hidden state h_{cls} can capture the target-context dependencies (Wang et al., 2020; Bai, Liu, & Zhang, 2021; Xing & Tsang, 2022e). Therefore, we adopt h_{cls} as the target-centric context semantics representation, which is used in Eq. 6.

5.1.1 NON-LOCAL SELF-ATTENTION

Although retrieving BERT’s self-attention matrix at the last layer seems an alternative way to obtain the self-attention graph, this approach cannot be used to construct the fully-connected semantic graph. The reason is that the Transformer blocks in BERT include many *multi-head* self-attention mechanisms, which segment word representations into multiple *local* subspaces. Therefore, the self-attention matrix corresponding to each head can only represent the *biased* word correlations in the *local* subspace. However, all of the subsequent modules work in the *global* semantic space.

To solve this, we employ a non-local self-attention layer after the BERT encoder so as to generate the *global* self-attention matrix $A^{sat} \in \mathbb{R}^{N_c \times N_c}$, which is the adjacent matrix of the semantic graph and represents the contextual word correlative information. Specifically,

2. CLS denotes classification. The $[\text{CLS}]$ token is added at the beginning of the input sequence and its representation is usually taken as the sentence-level representation for classification.

A^{sat} is obtained by:

$$A^{sat} = \text{Softmax} \left(\frac{(\hat{H}_c M_q)(\hat{H}_c M_k)^\top}{\sqrt{d}} \right) \quad (2)$$

The information aggregation is conducted along A^{sat} to update the hidden states:

$$H = A^{sat} \hat{H}_c M_v \quad (3)$$

where $M_* \in \mathbb{R}^{d \times d}$ are parameter matrices.

The target representation is obtained by applying average pooling over the target hidden states:

$$r^t = \frac{1}{N_t} \sum_i^{N_t} h_i^t \quad (4)$$

5.2 Heterogeneous Information Graph Construction

In this paper, we design the heterogeneous information graph (HIG) to integrate the background and property information of context target attributes, the syntactic information conveyed by the syntax graph, as well as the contextual word correlative information conveyed by the semantics graph. Fig. 3 illustrates the process of HIG construction. In the preprocessing stage, the syntax graph is automatically produced by the off-the-shelf dependency parser. Then we regard the syntax graph as an undirected graph, which is consistent with previous works (Zhang et al., 2019; Xing & Tsang, 2022e), and an example of its adjacent matrix A^{syn} is shown in Fig. 3. Note that in A^{syn} , we add the edges between the target words to let them fully connect to each other, aiming to enhance the target understanding.

In the training/testing process, the HIG is constructed based on the contextual target attributes, A^{syn} and A^{sat} . We first merge A^{syn} and A^{sat} with a coefficient α balancing the syntactic information and the contextual word correlative information:

$$\hat{A}_{i,j}^{hig} = \alpha \cdot A_{i,j}^{syn} + (1 - \alpha) \cdot A_{i,j}^{sat} \quad (5)$$

Then we add the nodes of contextual target attributes regarding the following rules:

- (1) There are edges between the attribute nodes and the target word nodes. This can enhance the target representation via integrating the background and property information contained in the attributes.
- (2) There are edges between the attribute nodes and the context word nodes whose corresponding words are connected to the target words in the syntax graph. Intuitively, the context words that are syntactically related to the target, can usually help to infer the target’s sentiment, and this is the reason that most recent models leverage the syntax graph. Therefore, integrating the background and property information contained in the attributes into the syntactically related context words can help to understand the context semantics more comprehensively, discovering more beneficial clues.
- (3) The above edges are unidirectional and directed from the attribute nodes to the target nodes or the context word nodes. The intuition behind this is that the information contained in the attribute nodes should be reserved, and they are supposed to provide the ‘pure’ information to other nodes. Therefore, HIG does not include the edges directed into the attribute nodes to guarantee that in the information aggregation process, they do not

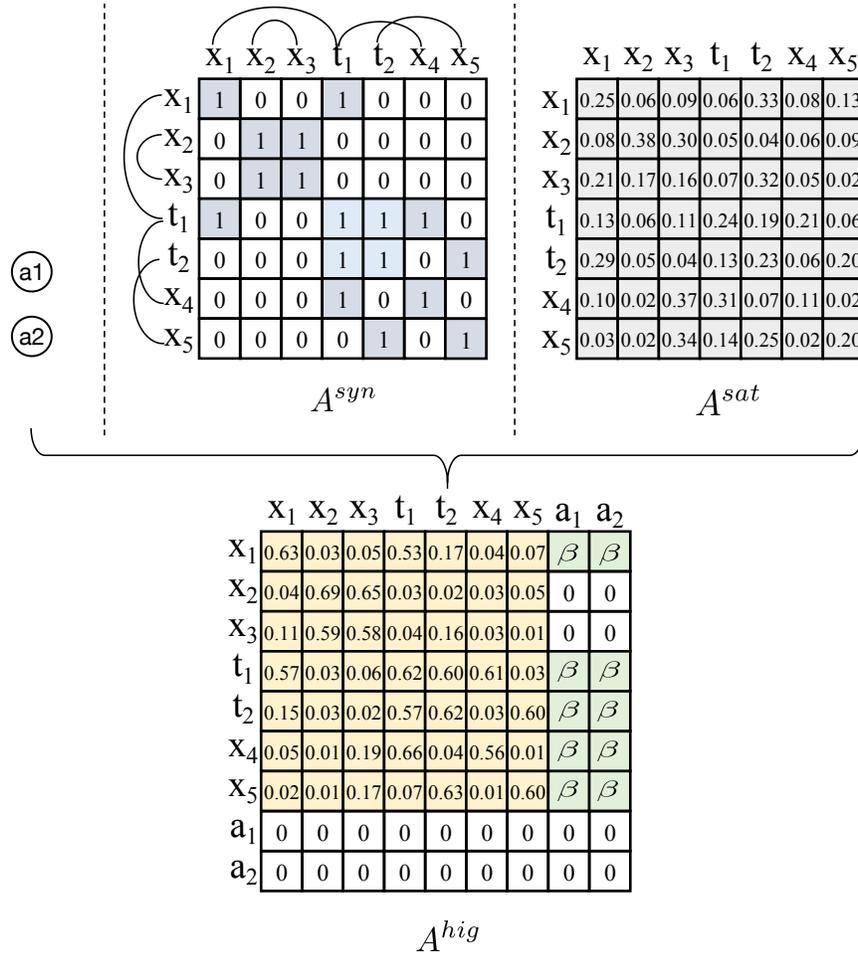


Figure 3: Illustration of the process of HIG construction. In this example, w.l.o.g, there are two contextual target attributes: a_1 and a_2 ; the coefficient α balancing the syntax graph and semantics graph is 0.5. And the arrows denote the edges/dependencies between the words on the syntax graph.

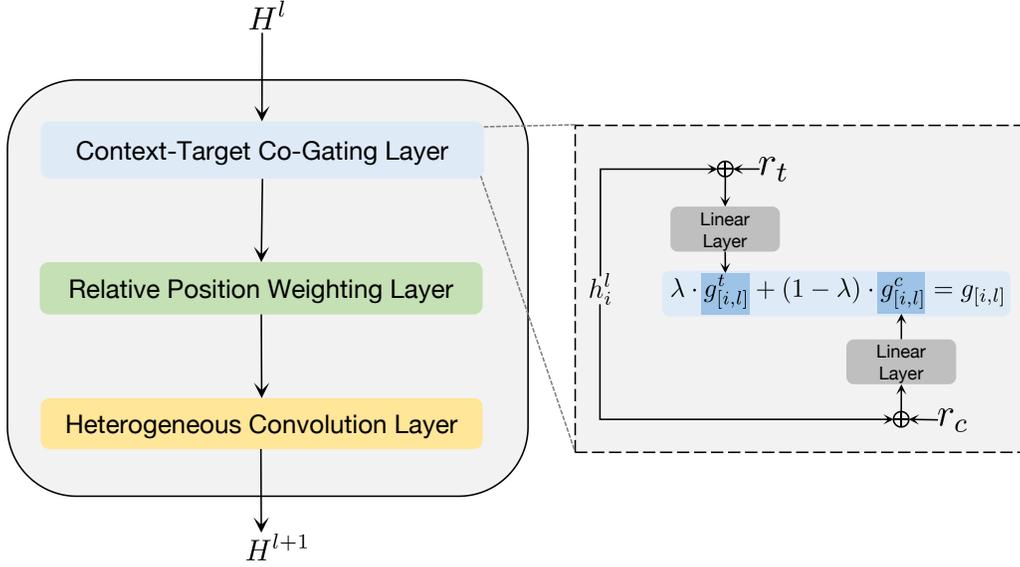
receive information from other nodes.

(4) There is a coefficient β to control the weight of the edges related to the attribute nodes. The edge weights between the context words are between 0 and 1. Intuitively, too large weights of the attribute nodes' edges would dilute the syntactic information and contextual word correlative information, while too small weights could not integrate enough background and property information of the target. Therefore, we use a hyper-parameter β to control the weight.

Then we can obtain the final HIG and its adjacent matrix A^{hig} . An example of A^{hig} is shown in Fig. 3.

5.3 Heterogeneous Information Gated Graph Convolutional Network

Fig. 4 illustrates the architecture of a single HIG²CN layer, which includes three sub-layers.

Figure 4: Illustration of a single HIG²CN layer.

5.3.1 CONTEXT-TARGET CO-GATING LAYER

We design a context-target co-gating layer to adaptively control the information flow regarding the target semantics and the target-centric context semantics. The objective is to distill beneficial information and filter out noisy information in the information aggregation process of HIG²CN. The inner details of the context-target co-gating layer are shown in Fig. 4. Specifically, it can be formulated as:

$$\begin{aligned}
 g_{[i,l]}^t &= W_g^t(h_i^l \oplus r_t) + b_g^t \\
 g_{[i,l]}^c &= W_g^c(h_i^l \oplus r_c^t) + b_g^c \\
 g_{[i,l]} &= \sigma\left(\lambda \cdot g_{[i,l]}^t + (1 - \lambda) \cdot g_{[i,l]}^c\right)
 \end{aligned} \tag{6}$$

where $g_{[i,l]}^t$, $g_{[i,l]}^c$ and $g_{[i,l]}$ are the target gating vector, context gating vector and the final co-gating vector, respectively; W_g^* and b_g^* denote the weight and bias, respectively; \oplus denotes vector concatenation operation; σ denotes the sigmoid function; λ denotes the hyper-parameter balancing the impacts of the target and context to the co-gating vector $g_{[i,l]}$; r_t denotes the target semantics, which is obtained in Eq. 4; r_c^t denotes the target-centric context semantics, and in this paper, it is h_{cls} , which has been obtained after encoding.

5.3.2 RELATIVE POSITION WEIGHTING LAYER

Intuitively, the target-related words are usually close to the target words in the review context. Therefore, we design a relative position weighting layer, which calculates the weight of each context word regarding its relative distance to the target words, aiming to

highlight potential target-related words. Specifically, the weight is calculated by:

$$w_p^j = 1 - \frac{\min_{\mu \leq \tau \leq \mu + N_t} |j - \tau|}{N_c + 1} \quad (7)$$

where μ denotes the first target word’s position in the context.

5.3.3 HETEROGENEOUS CONVOLUTION LAYER

In previous models, GCN (Kipf & Welling, 2017) is widely adopted for syntax graph modeling, which can be formulated as:

$$h_i^{l+1} = \text{ReLU} \left(\sum_{j=1}^{N_c} A_{ij} W_1^l h_j^{l-1} / (d_i + 1) + b^l \right) \quad (8)$$

where A denotes the adjacent matrix; W_1^l and b^l denote the weight and bias; d_i denotes the degree of node i .

However, it is inappropriate to adopt vanilla GCN to achieve the information aggregation on our HIG. On the one hand, it cannot take the co-gating vector $g_{[i,l]}$ and the relative position weight into consideration. On the other hand, it is intuitive that when a context node receives information, an attribute node’s information and another context node’s information have different contributions, while GCN cannot handle this. Therefore, in this paper, we propose the heterogeneous convolution layer to achieve the information aggregation in our HIG, and it can be formulated as:

$$\begin{aligned} h_i^{l+1} &= \text{ReLU} \left(\sum_{j=1}^{N_c} A_{ij}^{hig} W_1^l \hat{h}_j^l / (d_i + 1) + b^l \right) \\ \hat{h}_j^l &= g_{[j,l]} \odot (W^{\phi(j)} h_j^l) \cdot w^{\phi(j)} \\ W^{\phi(j)} &= \begin{cases} W^a & \phi(j) = \text{attribute} \\ I & \phi(j) \neq \text{attribute} \end{cases} \\ w^{\phi(j)} &= \begin{cases} 1 & \phi(j) = \text{attribute} \\ w_p^i & \phi(j) \neq \text{attribute} \end{cases} \end{aligned} \quad (9)$$

A^{hig} denotes the adjacent matrix; W_1^l and b^l denote the weight and bias; d_i denotes the degree of node i . \odot denotes the Hadamard product operation. $\phi(j)$ is a function that identifies whether node j is an attribute node. $W^{\phi(j)}$ is the transformation matrix that discriminates the attribute nodes and context nodes by projecting their representations into different spaces. For simplification, if node j is an attribute, we use a trainable weight matrix W^a to project its representation; if node j is not an attribute node, it is multiplied by the identity matrix I . $w^{\phi(j)}$ is the relative position weight of node j . If node j is an attribute node, it does not have the relative position weight because it is not included in the review context, so we set it to 1. Actually, the weight of the attribute node is controlled by the hyper-parameter β . If node j is not an attribute node, it has the relative position weight w_p^j that is calculated by Eq. 7.

By this means, HIG²CN can model the interactions among the background and property information of the target, the syntactic information and the contextual word correlative information. And in the information aggregation process, the information flow is adaptively controlled by the target’s semantics and the target-centric context’s semantics. After the stacked L layers of HIG²CN, the target word nodes are supposed to contain sufficient target sentiment clues. Then we obtain the final target representation by applying average pooling over all target word nodes:

$$R = \frac{1}{N_t} \sum_i^{N_t} h_{[i,L]}^t \quad (10)$$

5.4 Prediction and Training Objective

We first concatenate R with h_{cls} , and then the softmax classifier is used to produce the sentiment label distribution:

$$P = \text{Softmax}(\text{MLP}(R \oplus h_{cls})) \quad (11)$$

where MLP denotes the multi-layer perception layer.

Then we apply the argmax function to obtain the final predicted sentiment label:

$$\hat{y} = \text{argmax}_{k \in \mathcal{S}} P[k] \quad (12)$$

where \mathcal{S} denotes the set of sentiment classes.

For model training, the standard cross-entropy loss is employed as the objective function:

$$\mathcal{L} = - \sum_{i=1}^D \log(P[\mathcal{C}_i]) \quad (13)$$

where D denotes the number of training samples; \mathcal{C}_i denotes the index of the i -th training sample’s golden label.

6. Experiment

6.1 Benchmarks and Implementation Details

We conduct experiments on the Restaurant14, Laptop14 and Restaurant15 datasets (Pontiki, Galanis, Pavlopoulos, Papageorgiou, Androutsopoulos, & Manandhar, 2014; Pontiki, Galanis, Papageorgiou, Manandhar, & Androutsopoulos, 2015), which are widely adopted test beds for the TSC task. We pre-process the datasets following previous works (Zhang et al., 2019; Wang et al., 2020; Tian et al., 2021a). Table 2 lists the detailed statistics of the datasets.

We adopt the BERT_{base} uncased version³ for both encoding and the domain- and target-constrained cloze test. The BERT encoder is fine-tuned in the training process⁴. We adopt

3. Layer number is 12; hidden dimension is 768; attention head number is 12; total parameter number is 110M

4. The final loss of our model is back-propagated to not only HIG²CN’s parameters but also the loaded BERT’s parameters. In this way, BERT is fine-tuned to generate better contextual hidden states.

Dataset	Positive		Neutral		Negative	
	Train	Test	Train	Test	Train	Test
Laptop14	994	341	464	169	870	128
Restaurant14	2164	728	637	196	807	196
Restaurant15	912	326	36	34	256	182

Table 2: Dataset statistics.

the AdamW optimizer (Loshchilov & Hutter, 2019), which is used to train our model. We use the off-the-shelf dependency parser from the spaCy toolkit to obtain the syntax graph of the review context. Table 3 lists the details of the hyper-parameters.

Learning Rate	1e-5
Dropout Rate	0.3
Weight Decay Coefficient	0.05
Layer number of HIG ² CN	3
Dimension of Hidden State	768
Coefficient α	0.5
Coefficient β	0.5

Table 3: Dataset statistics.

Accuracy and Macro-F1 are used as evaluation metrics. Following previous works (Zhang et al., 2019; Tang et al., 2020; Bai et al., 2021), we report the average results over three runs with random initialization.

6.2 Baselines for Comparison

We compare our model with the following five groups of baselines. The models in the first four groups are fine-tuning-based and the ones in the last group are prompting-based.

(A) Neither BERT nor Syntax is leveraged:

1. IAN (Ma et al., 2017). It leverages the proposed interactive attention mechanism, which calculates the target words’ attention weights based on context semantics and the context words’ attention weights based on the target semantics.
2. RAM (Chen et al., 2017). It employs the recurrent attention mechanism that recurrently calculates the attention weights and aggregates the target-based semantics to the cell state.

(B) BERT is not leveraged while Syntax is leveraged:

3. ASGCN (Zhang et al., 2019). It applies multi-layer GCN over a syntax graph to capture syntactic information.
4. BiGCN (Zhang & Qian, 2020). It applies the GCN over the constructed hierarchical syntactic and lexical graphs.

(C) BERT is leveraged while Syntax is not leveraged:

5. BERT-SPC (Devlin et al., 2019). Taking the concatenated context-target pair as input, this model uses the output hidden state of the [CLS] token for classification.
6. AEN-BERT (Song, Wang, Jiang, Liu, & Rao, 2019). It stacks the attention layers in a multi-layer manner to learn deep target-context interactions.

(D) Both BERT and Syntax are leveraged:

7. ASGCN+BERT (Zhang et al., 2019). Based on ASGCN, we replace the original LSTM encoder with the same BERT encoder used in our model.
8. KGCapsAN-BERT (Zhang, Li, Xu, Leung, Chen, & Ye, 2020). It leverages different kinds of knowledge to enhance the capsule attention, and the syntactic knowledge is obtained by applying GCN over the syntax graph.
9. R-GAT+BERT (Wang et al., 2020). It captures the global relational information by considering both the target-context correlation and syntax relations.
10. DGEDT-BERT (Tang et al., 2020). It models the interactions between flat textual semantics and syntactic information through the proposed dual-transformer network.
11. RGAT-BERT (Bai et al., 2021). It leverages the dependency types between the context words to capture comprehensive syntactic information.
12. A-KVMN+BERT (Tian, Chen, & Song, 2021b). It leverages both word-word correlations and their syntax dependencies through the proposed key-value memory network.
13. BERT+T-GCN (Tian et al., 2021a). It models the dependency types among the context words by the proposed T-GCN and obtains a comprehensive representation via layer-level attention and attentive layer ensembling.
14. DualGCN+BERT (Li et al., 2021). It models the interactions between semantics and syntax by the proposed orthogonal and differential regularizers.
15. RAM+AAGCN+AABERT3 (Xing & Tsang, 2022e); This model augments RAM with the aspect-aware BERT and aspect-aware GCN to enhance the aggregation of target-related semantics.
16. dotGCN (Chen et al., 2022). This model employs a discrete latent opinion tree to augment the explicit dependency trees.

(E) Prompting-based Models:

17. BERT NSP Original and BERT LM Original (Seoh et al., 2021). These two models explore language modeling (LM) prompts and natural language inference (NLI) prompts, respectively.
18. AS-Prompt (Li et al., 2022). This model adopts continuous prompts to transfer the sentiment classification task into Masked Language Modeling (MLM) by designing appropriate prompts and searching for the ideal expression of prompts in continuous space.

In all of the above baselines, the BERT encoders are the same as ours, adopting the BERT-base uncased version. For fair comparisons, following Bai et al., 2021, we do not include the baselines that exploit external resources such as auxiliary sentences (Sun, Huang, & Qiu, 2019), extra corpus (Xu, Liu, Shu, & Yu, 2019) and knowledge base (Xing & Tsang, 2022f; Islam & Bhattacharya, 2022). Besides, since R-GAT+BERT, RGAT-BERT, A-KVMN+BERT, BERT+T-GCN and DualGCN+BERT did not report the average results in their original papers, we reproduce their average results on three random runs.

6.3 Main Result

The result comparison of our BERT+HIG²CN and the five groups of baselines on the three datasets are shown in Table 4. We can observe that BERT can significantly boost performance, and the basic model BERT-SPC obviously overpasses all LSTM-based models. Nevertheless, although BERT can effectively capture contextual knowledge, augmenting

Table 4: Comparison of evaluation results on the three benchmark datasets (in %). [†] indicates the results are reproduced using the official source code. [‡] denotes that the results were reported by Zhang et al., 2019. The best scores are in **bold** and the second best scores are underlined. Our model significantly outperforms baselines on all datasets and all metrics ($p < 0.05$ under t-test). For the t-test of our model and dotGCN on average F1, the calculated t-statistic is 5.57 and the degrees of freedom are 4.

Models	Laptop14		Restaurant14		Restaurant15		Average	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
IAN [‡]	72.05	67.38	79.26	70.09	78.54	52.65	76.62	63.37
RAM [‡]	74.49	71.35	80.23	70.80	79.30	60.49	78.01	67.55
ASGCN	75.55	71.05	80.77	72.02	79.89	61.89	78.74	68.32
BiGCN	74.59	71.84	81.97	73.48	81.16	64.79	79.24	70.04
BERT-SPC	78.47	73.67	84.94	78.00	83.40	65.00	82.27	72.22
AEN-BERT	79.93	76.31	83.12	73.76	-	-	-	-
ASGCN+BERT [†]	78.92	74.35	85.87	79.32	83.85	68.73	82.88	74.13
KGCapsAN-BERT	79.47	76.61	85.36	79.00	-	-	-	-
R-GAT+BERT [†]	79.31	75.40	86.10	80.04	83.95	69.47	83.12	74.97
DGEDT-BERT	79.8	75.6	86.3	80.0	84.0	71.0	83.37	75.53
A-KVMN+BERT [†]	79.20	75.76	85.89	78.29	83.89	67.88	82.99	73.98
BERT+T-GCN [†]	80.56	76.95	85.95	79.40	84.81	71.09	83.65	75.81
DualGCN+BERT [†]	80.83	77.35	86.64	80.76	84.69	71.58	84.05	76.56
RGAT-BERT	80.94	<u>78.20</u>	86.68	<u>80.92</u>	-	-	-	-
RAM+AAGCN+AABERT3	81.20	78.01	<u>86.79</u>	80.69	-	-	-	-
dotGCN	81.30	78.10	86.16	80.49	<u>85.24</u>	<u>72.74</u>	<u>84.23</u>	<u>77.11</u>
BERT NSP Original	<u>84.77</u>	76.93	77.96	73.24	-	-	-	-
BERT LM Original	84.26	75.90	77.61	73.06	-	-	-	-
AS-Prompt	85.64	76.71	78.37	74.19	-	-	-	-
BERT+HIG ² CN (ours)	82.18	78.70	87.50	82.05	86.53	75.71	85.40	78.82
standard deviation	±0.35	±0.49	±0.28	±0.47	±0.52	±0.69	-	-

BERT with syntactic information can bring further significant improvement. Although our model is based on the same BERT+Syntax+Attention framework as the existing best models, it obtains consistent improvements over them on both Acc and F1. Compared with the up-to-date best model dotGCN, in terms of Acc, our model gains improvements of 1.1%, 1.6%, and 1.5% on the Laptop14, Restaurant14, and Restaurant15 datasets, respectively. Additionally, in terms of F1, our model gains improvements of 0.8%, 1.9%, and 4.1% on the Laptop14, Restaurant14, and Restaurant15 datasets, respectively. As for the average performance, our model surpasses dotGCN by 1.4% and 2.2% in terms of Acc and F1, respectively. We can find that the prompting-based models usually obtain worse results compared with the BERT+Syntax+Attention models. We suspect the main reason is that although prompting-based models can make the best use of the language modeling ability, they cannot explicitly model the target-context interactions, which is crucial for TSC. Compared with prompting-based models, our model achieves consistent and significant improvements. The promising results of our model can be attributed to the fact that we exploit the contextual target attributes from BERT via the domain- and

target-constrained cloze test, and then effectively leverage them for comprehensive target and context understanding. Note that all baselines utilize the attention modules to extract the target-related sentiment information, while there is no attention module in our model. Therefore, the promising performance of our model proves that our HIG²CN can effectively and adaptively capture the target-related sentiment information via modeling the heterogeneous information interactions among the target’s background and property information in the attribute, the syntactic information, and the contextual word correlative information.

6.4 Ablation Study

Variants	Laptop14		Restaurant14		Restaurant15	
	Acc	F1	Acc	F1	Acc	F1
Full Model	82.18	78.70	87.50	82.05	86.53	75.71
NoAttribute	81.05	77.34	86.48	80.65	85.06	71.51
NoSyntax	81.33	77.69	86.57	80.56	85.24	71.74
NoSemantics	81.70	78.19	86.85	81.26	85.67	73.05
NoHIG ² CN	81.03	77.44	86.63	80.81	84.87	71.57
NoContextGating	81.59	77.97	86.93	81.34	85.98	73.70
NoTargetGating	81.78	77.84	87.02	81.16	86.04	73.82

Table 5: Results of ablation experiments. Our full model significantly outperforms the ablated variants on all datasets and all metrics ($p < 0.05$ under t-test).

There are two cores of our model: HIG and HIG²CN. To verify the necessity of their modules, we conduct two groups of ablation experiments, whose results are shown in Table 5.

HIG There are three sources of HIG: contextual target attributes, syntax graph and semantics graph. To study their effects, we design three variants: *NoAttribute*, *NoSyntax* and *NoSemantics*, respectively. *NoAttribute* is implemented by removing the attribute nodes on HIG. *NoSyntax* is implemented by directly adding the attribute nodes to the semantics graph. *NoSemantics* is implemented by directly adding the attribute nodes to the syntax graph. From Table 5, it can be seen that without attributes, the performance drops significantly, indicating that integrating the contextual target attributes to leverage the target’s background and property information can benefit the model. Another observation is that leaving out the syntax graph also leads to worse performance, proving that leveraging syntactic information is indispensable for the model because it can effectively enhance the context understanding to capture the target-related information. Moreover, we can find that if the semantics graph is not considered, the performance also drops sharply. This is because the semantics graph includes the correlation scores between every two words, which can help to capture more comprehensive contextual information.

HIG²CN To verify the effectiveness of HIG²CN, we design a variant *NoHIG²CN* via replacing HIG²CN with vanilla GCN. From Table 5, we can observe that *NoHIG²CN* obtains dramatic performance degradation, proving that HIG²CN is crucial for the whole model to capture the beneficial clues for TSC via heterogeneous information interactions. This can be

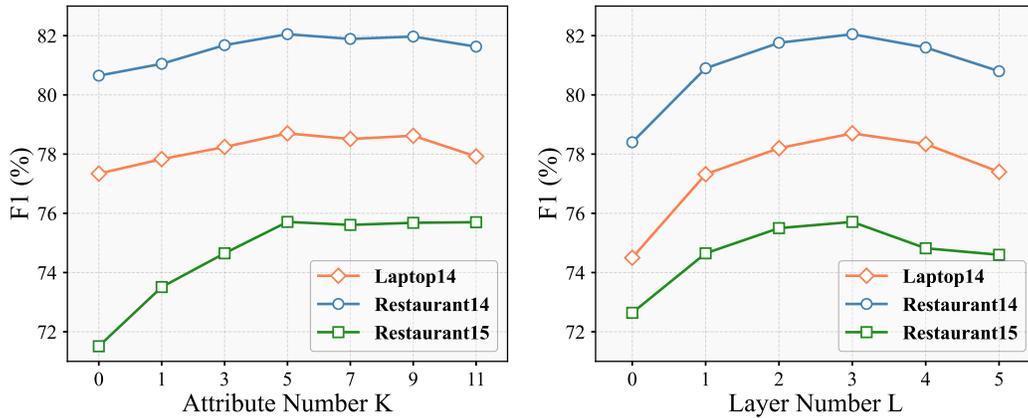


Figure 5: F1 score comparison with different contextual target attribute number (K) and HIG²CN layer number (L).

attributed to the fact that (1) the target-context co-gating layer in HIG²CN can adaptively control the information flow regarding the target and the target-centric context semantics; (2) HIG²CN can achieve more appropriate information aggregation via discriminating the information from attribute nodes and context nodes. To investigate the target-context co-gating layer, we design two variants: *NoContextGating* and *NoTargetGating*, which are implemented by removing the context gating vector $g_{[i,l]}^c$ and the target gating vector $g_{[i,l]}^t$, respectively. Compared with the full model, both *NoContextGating* and *NoTargetGating* obtain worse results. This proves that both the target semantics and the target-centric context semantics can provide indicating information for extracting beneficial information and filtering out noisy information in the information aggregation process.

6.5 Effect of Attribute Number

To study the effect of attribute number, we vary its value in the set of $\{0,1,3,5,7,9,11\}$, and the results (in F1) are shown in Fig. 5. We can observe that the results first increase and then stop rising or even drop when $K > 5$. We suppose there are two reasons. First, large K results in some low-confidence attributes containing noisy information. Besides, too many attributes lead to massive attribute information in HIG²CN, and then dilute the beneficial syntactic information and the contextual word correlative information.

6.6 Effect of HIG²CN Layer Number

We investigate the effect of HIG²CN layer number by setting it with different values ranging from 0 to 5, and the results are shown in Fig. 5. On all datasets, our model achieves the best result when $L = 3$. And with L increasing from 0 to 5, the performance first increases and then drops. This indicates that although more layers of HIG²CN can learn deeper interactions of the target’s background and property information, the syntactic information, and the contextual word correlative information, too large L leads to inferior performance since it causes the over-smoothing and over-fitting problems.

6.7 Case Study

Review	Context Target Attributes	Full Model	NoAttribute
1. Boot time is super fast , around anywhere from 35 seconds to 1 minute .	boot, speed, startup, drive, access	Positive (✓)	Neutral (×)
2. Entrees include classics like lasagna , fettuccine alfredo and chicken parmigiana.	pasta, spaghetti, tomato, pizza, salad	Neutral (✓)	Positive (×)

Table 6: Case study. The target words are in **bold**. The third and fourth columns denote the sentiment predicted by our full model and the NoAttribute variant, respectively. Sample 1 is from the Laptop14 test set (including 341 test samples in the positive class). Sample 2 is from the Restaurant14 test set (including 196 test samples in the neutral class).

To understand how context target attributes benefit TSC, we present a case study with two test samples, as shown in Table 6. In the first sample, without attributes, *NoAttribute* predicts the neutral sentiment regarding the semantics of ‘*time is fast*’. Augmented with the attributes (e.g., speed, startup), our model can give the correct prediction regarding the semantics of ‘*speed/startup is fast*’, which is generally positive. In the second sample, *NoAttribute* may hardly understand the target ‘fettuccine alfredo’, then predicts the positive sentiment via wrongly regarding ‘*like*’. Given the attributes, our model is aware that this sentence just introduces the different foods included in the entrees without expressing sentiment. Then it can predict the correct sentiment.

7. Discussion

Essentially, our approach is to retrieve the target-based contextual knowledge from the PTLM and then integrate this knowledge or target attribute information in the reasoning process. Our approach can be generalized into not only target-based tasks but also more general tasks. In Fig. 6, we show some tasks and instructions on how to apply our paradigm to the tasks.

For the tasks that have given targets (that is, Targeted Aspect-based Sentiment Classification, Conversational Aspect Sentiment Analysis), the first step is to adopt the domain- and target-constrained cloze test to generate the target-related candidate words that potentially provide the target’s property and background information. In this paper, we refer to these generated words as contextual target attributes since they are generated regarding the review context and contain the target’s attribute information. The second step is to design specific interaction graphs to integrate the beneficial attribute information into heterogeneous information interactions.

For more general tasks that do not have given targets (that is, spoken language understanding, Dialog Sentiment Classification and Act Recognition), we first propose to use TF-IDF to extract the topic/key words, which can be regarded as ‘targets’. Then the domain- and target-constrained cloze test can be conducted to obtain the contextual target attributes, whose beneficial attribute information is then integrated into heterogeneous information interactions. More specifically, in dialog sentiment classification and act recognition, we can use the extracted ‘targets’ to model the topic transition in the dialog and

Task Name	How to apply our paradigm
Targeted Aspect-based Sentiment Classification	Adopt the domain-and target-constrained cloze test to obtain the contextual target attributes, whose beneficial attribute information is then integrated into heterogeneous information interactions.
Conversational Aspect Sentiment Analysis	
Spoken Language Understanding	Use TF-IDF to retrieve the key words, which are regarded as targets. Then adopt the domain-and target-constrained cloze test to obtain the contextual target attributes, whose beneficial attribute information is then integrated into heterogeneous information interactions.
Dialog Sentiment Classification and Act Recognition	

Figure 6: The first two tasks are representatives of target-based tasks, which have given targets. The second two tasks are representatives of general tasks that do not have given targets. For these general tasks, we propose to use TF-IDF to retrieve the topic/key words, which are taken as targets. And then our method can be applied.

the relation between the mentioned topics and speakers. We believe this information can provide some beneficial knowledge that PTLM cannot provide.

8. Conclusion and Future Work

In this work, we propose a new perspective to leverage the power of pre-trained language models for TSC: contextual target attributes, which are generated by our designed domain-and target-constrained cloze test. To exploit the attribute for TSC, we propose a heterogeneous information graph and a heterogeneous information gated graph convolutional network to capture the target sentiment clues via modeling the interactions between the target’s background and property information, the syntactic information, and the contextual word correlative information. Our method makes the first attempt to simultaneously leverage the merit of both masked language modeling and explicit target-context interactions. Experiments show that we achieve new state-of-the-art performance.

Our approach makes the first attempt to simultaneously leverage the merits of masked language modeling (sufficient contextual knowledge) and fine-tuning method (explicit interactions achieved by upper-layer modules). Designed for TSC task, our approach has achieved significant improvements. In addition, our method provides further insights and can be generalized to other target-based or more general tasks in the future. For example, in dialog understanding tasks, we can use TF-IDF to extract the topic/key words which are regarded as targets. Then we can design some interaction graphs and corresponding networks to leverage these contextual extracted target attributes to model the topic transition in the dialog and the relation between the mentioned topics and speakers. This information

is the potential to improve these tasks because it provides some beneficial knowledge that the PTLM cannot provide.

Acknowledgements

This work was supported by National Key Research and Development Project of China (No. 2022YFC3502303) and Australian Research Council Grant (No. DP200101328). Ivor W. Tsang was also supported by A*STAR Centre for Frontier AI Research.

References

- Bai, X., Liu, P., & Zhang, Y. (2021). Investigating typed syntactic dependencies for targeted sentiment classification using graph attention neural network. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29, 503–514.
- Chen, C., Teng, Z., Wang, Z., & Zhang, Y. (2022). 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)*, pp. 2051–2064, Dublin, Ireland. Association for Computational Linguistics.
- Chen, P., Sun, Z., Bing, L., & Yang, W. (2017). Recurrent attention network on memory for aspect sentiment analysis. In *EMNLP*, pp. 452–461.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pp. 4171–4186.
- Huang, L., Sun, X., Li, S., Zhang, L., & Wang, H. (2020). Syntax-aware graph attention network for aspect-level sentiment classification. In *COLING*, pp. 799–810.
- Islam, S. M., & Bhattacharya, S. (2022). Ar-bert: Aspect-relation enhanced aspect-level sentiment classification with multi-modal explanations. In *Proceedings of the ACM Web Conference 2022*, pp. 987–998.
- Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *ICLR*.
- Li, G., Lin, F., Chen, W., Dong, D., & Liu, B. (2022). Prompt-based learning for aspect-level sentiment classification. In *Neural Information Processing: 29th International Conference, ICONIP 2022, Virtual Event, November 22–26, 2022, Proceedings, Part III*, pp. 509–520.
- Li, R., Chen, H., Feng, F., Ma, Z., Wang, X., & Hovy, E. (2021). Dual graph convolutional networks for aspect-based sentiment analysis. In *ACL*, pp. 6319–6329.
- Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9), 1–35.
- Loshchilov, I., & Hutter, F. (2019). Decoupled weight decay regularization. In *ICLR*.
- Ma, D., Li, S., Zhang, X., & Wang, H. (2017). Interactive attention networks for aspect-level sentiment classification. In *IJCAI*, pp. 4068–4074.

- Pontiki, M., Galanis, D., Papageorgiou, H., Manandhar, S., & Androutsopoulos, I. (2015). SemEval-2015 task 12: Aspect based sentiment analysis. In *SemEval 2015*, pp. 486–495.
- Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., & Manandhar, S. (2014). SemEval-2014 task 4: Aspect based sentiment analysis. In *SemEval 2014*, pp. 27–35.
- Seoh, R., Birlle, I., Tak, M., Chang, H.-S., Pinette, B., & Hough, A. (2021). Open aspect target sentiment classification with natural language prompts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 6311–6322.
- Song, Y., Wang, J., Jiang, T., Liu, Z., & Rao, Y. (2019). Attentional encoder network for targeted sentiment classification.. *CoRR*, *abs/1902.09314*.
- Sun, C., Huang, L., & Qiu, X. (2019). Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. In *NAACL*, pp. 380–385, Minneapolis, Minnesota.
- Tang, D., Qin, B., Feng, X., & Liu, T. (2016). Effective LSTMs for target-dependent sentiment classification. In *COLING 2016*, pp. 3298–3307, Osaka, Japan.
- Tang, H., Ji, D., Li, C., & Zhou, Q. (2020). Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. In *ACL*, pp. 6578–6588.
- Tang, J., Lu, Z., Su, J., Ge, Y., Song, L., Sun, L., & Luo, J. (2019). Progressive self-supervised attention learning for aspect-level sentiment analysis. In *ACL*, pp. 557–566.
- Tian, Y., Chen, G., & Song, Y. (2021a). Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble. In *NAACL*, pp. 2910–2922.
- Tian, Y., Chen, G., & Song, Y. (2021b). Enhancing aspect-level sentiment analysis with word dependencies. In *EACL*, pp. 3726–3739.
- Wang, K., Shen, W., Yang, Y., Quan, X., & Wang, R. (2020). Relational graph attention network for aspect-based sentiment analysis. In *ACL*, pp. 3229–3238.
- Wang, W., Pan, S. J., Dahlmeier, D., & Xiao, X. (2016a). Recursive neural conditional random fields for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 616–626, Austin, Texas. Association for Computational Linguistics.
- Wang, Y., Huang, M., Zhu, X., & Zhao, L. (2016b). Attention-based LSTM for aspect-level sentiment classification. In *EMNLP*, pp. 606–615, Austin, Texas.
- Xing, B., Liao, L., Song, D., Wang, J., Zhang, F., Wang, Z., & Huang, H. (2019). Earlier attention? aspect-aware LSTM for aspect-based sentiment analysis. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pp. 5313–5319. ijcai.org.
- Xing, B., & Tsang, I. (2022a). Co-guiding net: Achieving mutual guidances between multiple intent detection and slot filling via heterogeneous semantics-label graphs. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*,

- pp. 159–169, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Xing, B., & Tsang, I. (2022b). Darer: Dual-task temporal relational recurrent reasoning network for joint dialog sentiment classification and act recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 3611–3621.
- Xing, B., & Tsang, I. (2022c). Dignet: Digging clues from local-global interactive graph for aspect-level sentiment classification. *arXiv, 2201.00989*.
- Xing, B., & Tsang, I. (2022d). Group is better than individual: Exploiting label topologies and label relations for joint multiple intent detection and slot filling. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 3964–3975, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Xing, B., & Tsang, I. W. (2022e). Out of context: A new clue for context modeling of aspect-based sentiment analysis. *Journal of Artificial Intelligence Research, 74*, 627–659.
- Xing, B., & Tsang, I. W. (2022f). Understand me, if you refer to aspect knowledge: Knowledge-aware gated recurrent memory network. *IEEE Transactions on Emerging Topics in Computational Intelligence, 6*(5), 1092–1102.
- Xing, B., & Tsang, I. W. (2023a). Co-evolving graph reasoning network for emotion-cause pair extraction. In Koutra, D., Plant, C., Gomez Rodriguez, M., Baralis, E., & Bonchi, F. (Eds.), *Machine Learning and Knowledge Discovery in Databases: Research Track*, pp. 305–322, Cham. Springer Nature Switzerland.
- Xing, B., & Tsang, I. W. (2023b). Co-guiding for multi-intent spoken language understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 1*, 1–17.
- Xing, B., & Tsang, I. W. (2023c). Relational temporal graph reasoning for dual-task dialogue language understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 45*(11), 13170–13184.
- Xu, H., Liu, B., Shu, L., & Yu, P. (2019). BERT post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2324–2335, Minneapolis, Minnesota. Association for Computational Linguistics.
- Zhang, B., Li, X., Xu, X., Leung, K.-C., Chen, Z., & Ye, Y. (2020). Knowledge guided capsule attention network for aspect-based sentiment analysis. *IEEE ACM Trans. Audio Speech Lang. Process., 28*, 2538–2551.
- Zhang, C., Li, Q., & Song, D. (2019). Aspect-based sentiment classification with aspect-specific graph convolutional networks. In *EMNLP*, pp. 4568–4578.
- Zhang, M., & Qian, T. (2020). Convolution over hierarchical syntactic and lexical graphs for aspect level sentiment analysis. In *EMNLP*, pp. 3540–3549, Online.