Embedding Refinement Framework for Targeted Aspect-based Sentiment Analysis

Bin Liang, Rongdi Yin, Jiachen Du, Lin Gui, Yulan He, Min Yang, and Ruifeng Xu*, *Member, IEEE*

Abstract—The state-of-the-art approaches to Targeted Aspect-Based Sentiment Analysis (TABSA) are mostly built on deep neural networks with attention mechanisms. One problem is that embeddings of targets and aspects are either pre-trained from large external corpora or randomly initialized. We argue that affective commonsense knowledge and words indicative of sentiment could be used to learn better target and aspect embeddings. We therefore propose an embedding refinement framework called **RAEC** (**R**efining **A**ffective **E**mbedding from **C**ontext), in which sentiment concepts extracted from affective commonsense knowledge and word relative location information are incorporated to derive context-affective embeddings. Furthermore, a sparse coefficient vector is exploited in refining the embeddings of targets and aspects separately. In this way, embeddings of targets and aspects can capture the highly relevant affective words. Experimental results on two benchmark datasets show that our framework can be easily integrated with existing embedding-based TABSA models and achieves state-of-the-art results compared to models relying on pre-trained word embeddings or built on other embedding refinement methods.

Index Terms—Target Sentiment Analysis, Aspect Sentiment Analysis, Embedding Refinement, Affective Knowledge

1 INTRODUCTION

Sentiment analysis or opinion mining is one of the key tasks in natural language processing (NLP). It aims to infer polarities and/or retrieve opinions from text [2], [3]. While coarse-grained sentiment analysis detects sentiment labels at the sentence or document level, fine-grained sentiment analysis aims to reveal the polarity towards a specific target or aspect [4], [5], [6]. It is a more challenging task compared to sentence- or document-level sentiment analysis since it needs to leverage the affective information from both context and the specific targets/aspects in order to detect the target- or aspect-dependent polarity. Fine-grained sentiment analysis can be further classified as aspect-based sentiment analysis (ABSA) [7], target-dependent sentiment analysis (TDSA) [8] and targeted aspect-based sentiment analysis (TABSA) [9]. For example, in sentence "*Living in London is good but very expensive*", "*London*" is the target contained in the sentence (corresponding to aspect "*LIVE*" and "*PRICE*"). TDSA needs to detect sentiment polarity for target "*London*".

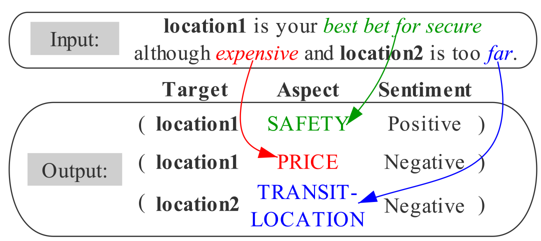


Fig. 1. Example of an input sentence and output labels in TABSA task. Affective words and corresponding aspects are in the same color.

While the goal of ABSA is to detect sentiment polarities for different aspects mentioned in the sentence, i.e. "*LIVE*" and "*PRICE*". TABSA aims at detecting aspects according to the specific target and inferring sentiment polarities correspondingly for different target-aspect pairs simultaneously. Hence, it requires the extraction of the contextual sentiment features for both targets and aspects [10], [11]. Here, the difference between ABSA and TABSA is that ABSA aims to detect the set of tuples $\{(aspect, polarity)\}$ of the sentence, while the goal of TABSA is to detect the set of tuples $\{(target, aspect, polarity)\}$ of the sentence.

Consider a review of living locations in **Figure 1**, here entities are replaced by the placeholders `location1` and `location2`. These entities are considered as targets. For the target `location1`, the polarity is positive towards the aspect "`SAFETY`" but negative on aspect "`PRICE`", while for the target `location2`, there is a negative polarity to the aspect "`TRANSIT-LOCATION`". We can observe that even in the same sentence, the sentiment polarities can be different when considering different targets. We also notice that the sentiment is determined by the associated target-aspect pair. That is, the aspect and sentiment expressions for a given target are exhibited via contextual affective words which are highly relevant to the corresponding target and aspect.

• *B. Liang, R. Yin, J. Du are with the School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen 518055, China, Joint Lab of HITSZ and China Merchants Securities, Shenzhen 518055, China.*
E-mail: bin.liang@stu.hit.edu.cn, rongdiyin@gmail.com, Jacobvan199165@163.com;
• *G. Lin, Y. He are with the Department of Computer Science, University of Warwick, Coventry CV47AL, U.K.*
E-mail: {lin.gui, Yulan.He}@warwick.ac.uk;
• *M. Yang is with Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China.*
E-mail: min.yang@siat.ac.cn;
• *R. Xu is with the School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen 518055, China, Peng Cheng Laboratory, Shenzhen 518055, China.*
E-mail: xuruifeng@hit.edu.cn.
• *The work is the extended version of our previously published conference paper of ACL [1].*

* *Ruifeng Xu is the Corresponding Author.*

Word embeddings, which are usually learned from contextual semantic information, play an important role in many NLP tasks [12], [13], [14], [15], [16], [17], [18], [19], [20]. In sentiment analysis, word embeddings learned by neural models can capture certain types of sentiment features from input texts [21], [22], [23], [24], [25], [26], [27]. Moreover, many embedding refinement methods have been proposed which refine the pre-trained word embeddings based on external knowledge for a better modeling of sentiment information [17], [28], [29], [30], [31], [32], [33]. However, these methods only focus on learning sentiment embeddings, without considering the contextual relations between targets and aspects in the task of TABSA.

Recently, the attention mechanism has achieved a great success in many NLP tasks [34], [35], [36], [37], [38], [39], [40], [41], [42]. In the task of (T)ABSA, attention mechanism can help models effectively distinguishing the sentiment polarities of different aspects in the same sentence [9], [10], [11], [43], [44], [45], [46], [47], [48], [49]. But these methods usually ignore the interdependencies between targets and aspects in text and do not leverage the contextual-affective information with regards to specific targets and aspects.

To address the problems mentioned above, we propose a novel embedding refinement framework for TABSA, called **R**efining **A**ffective **E**mbeddings from **C**ontext (**RAEC**). More concretely, we first incorporate sentiment knowledge in pre-trained contextual word embeddings to obtain context-affective embeddings of words. Then, we refine the embeddings of targets and aspects from the context in order to capture the contextual interdependencies between aspects and their associated targets based on the aspect-specific sentiment polarity. In this way, the context-affective information can be incorporated into the embedding refinement framework to capture the affective relations for targets and aspects. The refined embeddings are then fed into a neural network-based TABSA model for the detection of the aspect-level sentiment polarity for a given target. The main contributions of our work can be summarized as follows:

- A novel deep learning framework for affective embedding refinement from context is proposed to learn better embeddings for targets, aspects and context. As a general framework, it can be directly incorporated in existing embedding-based methods to achieve better performance in the TABSA task.
- The proposed embedding refinement strategy essentially considers the interplay of targets, aspects, affective words and context and learns better target and aspect embeddings as will be shown in the experiments section.
- Substantial experiments have been conducted, in which the results on SentiHood and Sem15 show that our proposed framework can beat the state-of-the-art approaches.

The rest of the paper is organized as follows. Section 2 presents a survey of related work on target-based and aspect-based sentiment analysis. Section 3 describes our proposed framework for TABSA. Section 4 presents experimental setup and evaluation results. Finally, Section 5 concludes the paper and outlines the future directions.

2 Related Work

2.1 Target-dependent Sentiment Analysis

Target-dependent Sentiment Analysis (TDSA) aims at detecting the sentiment polarity of a given target. Many neural networks-based models achieve remarkable performance in TDSA [6], [8], [50], [51], [52], [53], [54]. Among them, an adaptive recursive neural network, which employs a novel adaptive multi-compositionality layer in a recursive neural network, was proposed to extract the sentiment information of words towards a target depending on the associated context and the syntactic structure [50]. To model the relatedness of a target word with its context words, a neural model built upon left and right LSTMs was proposed to select the relevant parts of context to infer the sentiment polarity towards a target [8]. To learn features better, a feature-enhanced multi-view co-attention network was developed to learn a better multi-view sentiment-aware and target-specific sentence representation [52]. A memory network that automatically learns the interactions among aspect words and opinion words was proposed to extract sentiment features of a given target or aspect. The model was extended in a multi-task setting to solve a fine-grained opinion mining problem, which involves the identification of aspects and opinion terms within each sentence and the simultaneous categorization of the identified terms [6]. More recently, built on BERT [55], several target-dependent models, which can extract the bidirectional contextual semantic dependencies for a given target, were proposed to improve the performance of target-dependent sentiment classification [54].

2.2 Aspect-Based Sentiment Analysis

Aspect-Based Sentiment Analysis (ABSA) aims at determining the sentiment polarities towards specific aspects in text. Motivated by the success of attention mechanisms, much recent work adopted attention-based neural models to leverage the aspect-specific sentiment information in text [43], [44], [45], [46], [47], [56], [57], [58], [59], [60], [61], [62], [63], [64]. For instance, an attention-based LSTM was proposed to pay attention to different key parts for different aspects in an input sentence, which can discriminate different sentiment polarities for different aspects [43]. A multi-grained attention network was proposed to capture the word-level interactions between aspects and context by fine-grained and coarse-grained attention mechanisms [47]. A multiple attention-based deep memory network, which focuses on both contextual and positional attentions for a given aspect, was used to explicitly capture the importance of each context word when inferring the sentiment polarity of an aspect in the task of ABSA [44]. Analogously, another memory-based model adopted the multiple-attention mechanism to capture sentiment features separated by a long distance, which is able to learn different tailor-made memories for different aspects. The weighted-memory mechanism provided a tailor-made memory for different opinion targets of a sentence [46]. A gated convolutional neural network (CNN) was utilized to selectively extract the sentiment features according to the given aspect through the gated operation [63]. An aspect-specific graph convolutional network (GCN) over the dependency tree of a sentence was exploited to extract the word dependencies for a specific aspect [60].

2.3 Targeted Aspect-based Sentiment Analysis

As a more challenging sentiment analysis task, Targeted Aspect-based Sentiment Analysis (TABSA) aims at determining the sentiment polarity for an aspect of a certain target. Hence both targets and aspects information needs to be extracted from text [1], [9]. Based on an input sentence and a given target, TABSA needs to leverage sentiment information from the contextual words to detect the associated aspect for a given target, and infer the sentiment polarity for each target-aspect pair. Most previous work usually tackled TABSA in two subtasks: aspect detection and sentiment classification for a specific aspect. For instance, a feature-based logistic regression model and two LSTM-based models were proposed together with SentiHood [9], which can extract important sentiment features and perform aspect-based sentiment analysis for each target-aspect pair. To leverage the commonsense knowledge of sentiment-related concepts, an attention-based LSTM [10], which utilized both the target-level and the sentence-level attentions to capture the important features in the context, was proposed to perform both aspect detection and aspect-based sentiment analysis. A recurrent entity network was proposed to independently track and update the states of targets at the right time by utilizing external memory chains with a delayed memory update mechanism for both aspect detection and sentiment analysis in the task of TABSA [11]. Several BERT-based models were proposed to construct an auxiliary sentence for the input sentence and transform (T)ABSA into a sentence-pair classification task [65].

2.4 Sentiment Embeddings

Pre-trained embeddings are helpful for many NLP tasks [18], [55], [66]. Various approaches have been proposed to learn low-dimensional dense word vectors, which can efficiently capture semantic and syntactic information for contextual words. To improve the representations of contextual words for sentiment analysis, several neural network-based models with tailoring loss functions were proposed to learn sentiment-specific word embeddings [67]. In addition, some embeddings refinement methods were proposed to generate better semantic word representations. For instance, a word vector refinement model to correct the pre-trained word embedding by using manifold learning was proposed to bring the similarity of words in the Euclidean space closer to word semantics [33]. To learn sentiment word embeddings, a sentiment embedding refinement model based on adjusting the representations of words was used in sentiment analysis to make the word embeddings closer to words which are semantically similar and bear the same polarities and further away from words with opposing polarities [31].

Existing embedding-based TABSA models typically only utilize pre-trained word embeddings. Further, these word embeddings cannot capture the affective information in text, such as the contextual sentiment information of both targets and aspects. Since the affective commonsense knowledge could be potentially important for sentiment analysis [23], [29], [68]. Some embedding refinement methods were proposed to learn sentiment word representations in the task of sentiment analysis. However, they usually only generate word embeddings containing sentiment information from the contextual semantics and do not refine the representations for the specific targets and aspects in the task of TABSA. To extract the relations between targets and aspects in the context, a context-aware embedding refinement method was proposed in TABSA, which can refine the targets and aspects embeddings based on the highly relavent words in context [1].

This paper is a significant extension of our preliminary work in the conference paper [1]. The differences between this paper and [1] are summarized as follows: 1) we utilize the affective commonsense knowledge and position encoding to refine the pre-trained word embedding representing the input sequence of words, enabling the capturing of the sentiment dependencies of the contextual words; 2) the sentiment polarity information is incorporated into the embedding refinement process of both targets and aspects, making them better encoding the relevant polarity information; 3) instead of using a simple step function, a precise function is adopted to compute the sparse coefficient vector in our **RAEC** framework, which controls the speed of sparsification of word embeddings and avoids the ignorance of important words.

3 Methodology

In this section, we describe the proposed method in details. The illustration of our embedding refinement framework is demonstrated in **Figure 2**.

3.1 Task Description

Let $s = \{w_1, w_2, \dots, t, \dots, w_n\}$ be a sequence of words of an input sentence, where t is a target. There might be one or multiple targets in a sentence. Here, the input sentence can be represented as an embedding matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$, where n is the sentence length, m is the dimension of word embeddings, i.e. each word can be represented as an m-dimensional embedding $\mathbf{x} \in \mathbb{R}^m$. The embeddings of the target $\mathbf{t} \in \mathbb{R}^m$ and the aspect $\mathbf{a} \in \mathbb{R}^m$ are an average of their constituting word embeddings if they are multi-word phrases. The word embedding of "*Positive*", "*Negative*" or "*Neutral*" is used to represent the sentiment polarity embedding $\mathbf{r} \in \mathbb{R}^m$, $r \in \{Positive, Negative, None/Neutral\}$. In TABSA, given a sentence s, a pre-identified set of targets T and a fixed set of aspects A, our goal is to detect the aspect $a \in A$ and then identify its associated sentiment polarity $r \in \{Positive, Negative, Neutral\}$ for each target-aspect pair (t, a). For example, in **Figure 1**, the polarity of `(location1, SAFETY)` is "*Positive*", while the polarity of `(location1, PRICE)` is "*Negative*", that is, the sentiment polarities may be opposite when considering different aspects in the same sentence.

3.2 Overview

Our **RAEC** framework consists of two parts: the contextual affective embeddings generation and the embeddings refinement for targets and aspects, as shown in **Figure 2**.

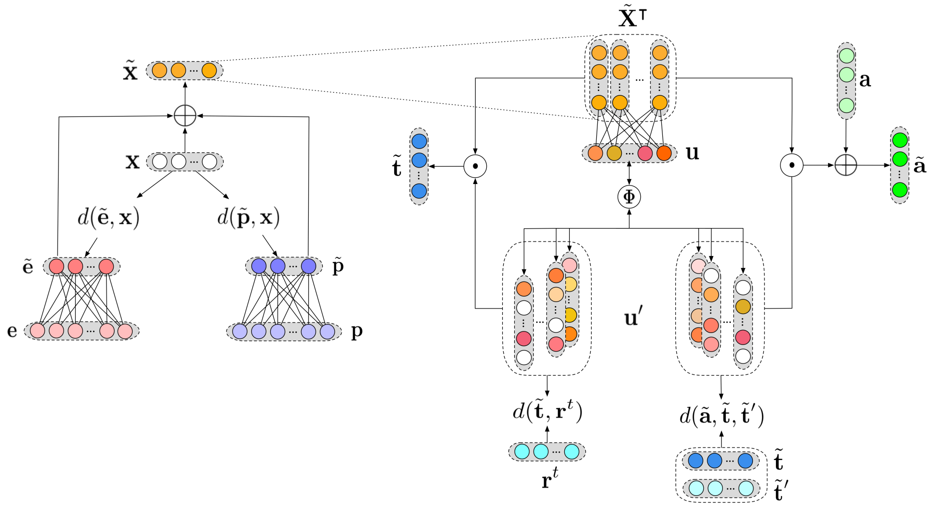


Fig. 2. The illustration of our RAEC framework. Φ is the step function, which is utilized to compute the sparse coefficient vector. $\cdot$ denotes matrix multiplication. $+$ denotes vector addition.

The part of the contextual affective embeddings generation mainly contains three components: 1) affective embeddings derivation, which integrates affective information into word embedding, 2) position encoding computation, which focuses the significant relative distance between words and the distinct target, and 3) context-affective embedding generation, which derives each contextual word representation based on the affective embedding and position encoding. The part of the embeddings refinement for targets and aspects mainly contains four components: 1) contextual embedding matrix input module, which takes the context-affective embeddings of the sentence as input, 2) important contextual words selection, which identifies the distinct important contextual words for the distinct target/aspect from the context, 3) target embedding refinement, which refines the target representation according to the corresponding selected contextual words, and 4) aspect embedding refinement, which refines the aspect representation based on the target and the corresponding selected contextual words.

Given an original embedding $\mathbf{x} \in \mathbb{R}^m$, an AffectiveSpace embedding $\mathbf{e} \in \mathbb{R}^d$, and a position encoding $\mathbf{p} \in \mathbb{R}^l$, a neural-based computation is first performed to generate the projected affective embedding $\tilde{\mathbf{e}} \in \mathbb{R}^m$ and the contextual position encoding $\tilde{\mathbf{p}} \in \mathbb{R}^m$. Then the projected affective embedding, the projected contextual position embedding and the word embedding are combined as a context-affective embedding $\tilde{\mathbf{x}} \in \mathbb{R}^m$ and fed into the neural refinement component. The hidden state $\mathbf{u}' \in \mathbb{R}^n$ of the refinement component is a sparse coefficient vector, which is utilized to compute the refined embeddings of the target $\tilde{\mathbf{t}} \in \mathbb{R}^m$ and the aspect $\tilde{\mathbf{a}} \in \mathbb{R}^m$. Afterwards, the refined embeddings of the target and the aspect are combined with context-affective word embeddings and fed into the embedding-based model to identify the polarity corresponding to the specific target-aspect pair.

3.3 Incorporating Affective Knowledge

Sentiment-based commonsense knowledge is potentially useful for improving the performance of sentiment classification. AffectiveSpace [69][1] is a commonsense knowledge resource that containing rich affective information of words. Here the AffectiveSpace maps the concepts in SenticNet [70] to continuous low-dimensional embeddings $\mathbf{e} \in \mathbb{R}^d$ without losing the information of semantic and affective relatedness in the original space. However, the low-dimensional embeddings from AffectiveSpace reside in a semantic space different from that of pre-trained word embeddings, thus making it difficult to combine these two for model learning in TABSA. To address this problem, we use a fully connected network to project the AffectiveSpace embeddings into a new semantic space and make the projected embeddings close to their corresponding pre-trained word embeddings as much as possible. First, the projection is performed by:

$$\tilde{\mathbf{e}} = f(\mathbf{W} \cdot \mathbf{e} + \mathbf{b}) \tag{1}$$

where f is a non-linear function such as sigmoid, $\mathbf{W} \in \mathbb{R}^{m \times d}$ and $\mathbf{b} \in \mathbb{R}^m$ denote the weight matrix and the bias, respectively. Next, for each word, the squared Euclidean distance between its AffectiveSpace embedding and its pre-trained word embedding is iteratively minimized. The objective function is defined as:

$$d(\tilde{\mathbf{e}}, \mathbf{x}) = \sum_{i=1}^{m} (\tilde{e}_i - x_i)^2 \tag{2}$$

Through the iterative procedure, the projected AffectiveSpace embedding $\tilde{\mathbf{e}} \in \mathbb{R}^m$ will be iteratively updated and get closer to its corresponding word embedding.

3.4 Incorporating Position Encoding

Intuitively, the position information may be different with respect to different targets even in the same sentence. That is, the relative positions of words are different with respect to different targets, thus exerting different influences on the targets. To this end, we inject position encoding into embedding learning. Inspired by [44], we define the position encoding, $\mathbf{p}^a \in \mathbb{R}^l$ and $\mathbf{p}^b \in \mathbb{R}^l$, as the relative distance

1. http://sentic.net/downloads

between word w_i and different targets (such as `location1` and `location2` in Figure 1). Each element in $\mathbf{p}^a$ or $\mathbf{p}^b$ is calculated as:

$$p_j^a = 1 - |pos - pos_1|/n \tag{3}$$
$$p_j^b = 1 - |pos - pos_2|/n \tag{4}$$

where pos is the position of word, pos_1 and pos_2 represent the positions of `location1` and `location2`, respectively. $|\cdot|$ is an absolute value function, and n is the sentence length. If there is only one target, the position encoding of other target is denoted as $\mathbf{p}^b = \mathbf{0}$.

To leverage the context information for both targets simultaneously, position encodings $\mathbf{p}^a \in \mathbb{R}^l$ and $\mathbf{p}^b \in \mathbb{R}^l$ are summed to get the associative position encoding $\mathbf{p} \in \mathbb{R}^l$ of the word:

$$\mathbf{p} = \mathbf{p}^a + \mathbf{p}^b \tag{5}$$

In accordance with incorporating affective knowledge, a fully connected network and the minimizing of the squared Euclidean distance are used to generate a contextual position embedding from both the position encoding and the word embedding[2]:

$$\tilde{\mathbf{p}} = f(\mathbf{W} \cdot \mathbf{p} + \mathbf{b}) \tag{6}$$
$$d(\tilde{\mathbf{p}}, \mathbf{x}) = \sum_{i=1}^{m} (\tilde{p}_i - x_i)^2 \tag{7}$$

where $\tilde{\mathbf{p}} \in \mathbb{R}^m$ is the contextual position embedding, $\mathbf{W} \in \mathbb{R}^{m \times l}$ and $\mathbf{b} \in \mathbb{R}^m$ denote the weight matrix and the bias.

To incorporate commonsense knowledge and position information into the model, the AffectiveSpace embedding $\tilde{\mathbf{e}} \in \mathbb{R}^m$, the contextual position embedding $\tilde{\mathbf{p}} \in \mathbb{R}^m$, and the word embedding $\mathbf{x} \in \mathbb{R}^m$ are combined to get the representation of context-affective embedding:

$$\tilde{\mathbf{x}} = \tilde{\mathbf{e}} + \tilde{\mathbf{p}} + \mathbf{x} \tag{8}$$

where $+$ is vector addition.

3.5 Embedding Refinement for Targets and Aspects

We describe the refinement of embeddings of targets and aspects in this subsection. The aim of refining target and aspect representations is to learn a contextual affective embedding for a target or an aspect via a sparse coefficient vector. In this way, a set of highly relevant words are extracted from context to represent the target and the aspect, which makes the target and aspect embeddings better associated with its corresponding sentiment polarity.

We feed the context-affective embedding matrix into a fully connected network whose output is passed to a step function to generate the sparse coefficient vector:

$$\mathbf{u} = f(\tilde{\mathbf{X}}^{\intercal} \cdot \mathbf{W} + \mathbf{b}) \tag{9}$$
$$\mathbf{u}' = \mathbf{\Phi}(\mathbf{u}) \tag{10}$$

where $\mathbf{W} \in \mathbb{R}^{m \times 1}$ and $\mathbf{b} \in \mathbb{R}^1$ denote the weight and the bias respectively, Φ is a step function given a real value:

$$\Phi(u_i) = \begin{cases} u_i & j = 1 \\ u_i & 1 < j < n \text{ and } u_i \geqslant mean(\dot{\mathbf{u}}[:j]) \\ u_i & j \geqslant n \text{ and } u_i \geqslant mean(\mathbf{u}) \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

where $mean(\cdot)$ is an average function, $\dot{\mathbf{u}}$ represents sorting the elements of $\mathbf{u}$ in ascending order, i.e. the value of index 0 is minimum. j denotes the number of iterations. In the process of iterations, we control the speed and degree of sparsification of elements in $\mathbf{u}$ to better capture the important context words and prevent the ignorance of highly relevant affective words. Afterwards, the target embedding is reconstructed from the context-affective embedding matrix according to the sparse coefficient vector defined as:

$$\tilde{\mathbf{t}} = \tilde{\mathbf{X}} \cdot \mathbf{u}' \tag{12}$$

where $\tilde{\mathbf{X}} \in \mathbb{R}^{m \times n}$ is the input context-affective embedding matrix composed by $\tilde{\mathbf{x}}$, $\tilde{\mathbf{t}}$ is the refined target representation.

For each target, the embedding refinement step aims to get a contextual affective embedding by iteratively minimizing the squared Euclidean distance between the aspect-sentiment representation of a given target and its highly related words in the sentence:

$$d(\tilde{\mathbf{t}}, \mathbf{r}^t) = \sum_{i=1}^{m} (\tilde{t}_i - r_i^t)^2 + \sum_{j=1}^{n} \lambda u'_j \tag{13}$$

where $\lambda u'_j$ aims to control the degree of sparseness of vector $\mathbf{u}'$ by limiting the element value. Here, 1) the sentiment expression of the target is associated with the corresponding aspect. 2) The target representations should be different in different polarities even though for the same aspect. Thus we consider both the information of aspect and the sentiment polarity when we refine the target representation, i.e. $\mathbf{r}^t$ is the representation of the corresponding aspect $\mathbf{a}$ and its associated sentiment polarity $\mathbf{r}$ for a given target[3]:

$$\mathbf{r}^t = \mathbf{r} + \mathbf{a} \tag{14}$$

Through the iterative process, the vector representation of a target will be iteratively updated until the number of the non-zero elements of vector $\mathbf{u}'$ is less than a threshold value: $k \leqslant c$, where k is the number of the non-zero elements of vector $\mathbf{u}'$ and c is the threshold value. That is, we extract up to c most relevant affective words for refinement. Thus in this way, the representation of the target associated with the corresponding aspect is distinct in the distinct context.

Intuitively, aspect words in the context would be helpful for aspect detection. For example, the word "*price*" is indicative to the aspect "`PRICE`". Analogous to embedding refinement of targets, aspect embeddings can be fine-tuned from the contextual affective embedding matrix of an input sentence. By considering both highly relevant words and the

2. We perform early stopping during training, i.e. we stop the training when the verification/test set loss of the current epoch is greater than that of the previous epoch. Hence the contextual position embedding can retain the original information and get closer to the word embedding simultaneously.

3. We use the training data to refine the representations in our work, thus the sentiment polarity of the instance is known. Additionally, to present the generalizability of our work to some general instances that refer to the implicit targets (such as the target entity is masked with "location1" or "location2" in SentiHood dataset), we do not inject target representation in the representation of refinement goal.

TABLE 1
Statistics of the experimental datasets.

Dataset	SentiHood			Sem15	
	Train	Dev	Test	Train	Test
Positive	1626	406	810	949	334
Negative	834	204	406	269	193
None	5556	1422	2700	4712	2093
Total	8016	2031	3916	5930	2620

aspect word embedding, each element in the refined aspect embedding $\tilde{\mathbf{a}}$ can be computed as follows:

$$\tilde{a}_i = a_i + \tilde{X}_i \cdot u'_i \tag{15}$$

Through iteratively minimizing the squared Euclidean distance, the refinement step aims at updating an aspect embedding such that it is close to the corresponding target as much as possible and is further away from other targets in the context. Here, the objective function of the aspect embedding refinement is:

$$d(\tilde{\mathbf{a}}, \tilde{\mathbf{t}}, \tilde{\mathbf{t}}') = \sum_{\tilde{\mathbf{t}}' \in \mathbf{T}, \tilde{\mathbf{t}}' \neq \tilde{\mathbf{t}}} \sum_{i=1}^{m} (\tilde{a}_i - \tilde{t}_i)^2 - (\tilde{a}_i - \tilde{t}'_i)^2 + \sum_{j=1}^{n} \lambda u'_j \tag{16}$$

where $\tilde{\mathbf{t}}$ is the embedding of the specific target and each $\tilde{\mathbf{t}}'$ is the irrelevant one in the context.

3.6 Learning Objective

Our **RAEC** framework can be easily integrated with embedding-based TABSA models for aspect detection and sentiment analysis. In this section, we describe the universal training objective of both aspect detection and sentiment analysis in TABSA. Both the contextual affective embedding matrix of a sentence and the refined embeddings of target and aspect are fed as input into the model (such as Delayed-memory [11]) to capture the hidden representation which will be passed to a softmax layer for aspect detection and sentiment classification:

$$\mathbf{v} = \textit{Delayed-memory}(\tilde{\mathbf{X}}, \tilde{\mathbf{a}}, \tilde{\mathbf{t}}) \tag{17}$$

After that, the output distribution $\mathbf{y}$ is predicted for a given target-aspect pair:

$$\mathbf{y} = \textit{softmax}(\mathbf{W} \cdot \mathbf{v} + b) \tag{18}$$

where $\mathbf{W}$ and b are the trainable parameters of *softmax*. The objective to train the classifier is defined as the minimization of the cross-entropy loss between predicted and ground-truth distributions:

$$\mathcal{L} = \textit{CrossEntropy}(\mathbf{y}, \hat{\mathbf{y}}) \tag{19}$$

where $\hat{\mathbf{y}}$ is the ground-truth output distribution of a target-aspect pair.

4 Experiments

4.1 Dataset

We conduct experiments on SentiHood [9] and the restaurant domain from Semeval 2015 [71]. The annotated sentences in SentiHood containing 1 or 2 targets corresponding to several aspects selected from 12 predefined aspects. There are 5,215 sentences in SentiHood, 3,862 sentences have single target and 1,353 sentences have multiple (two) targets. Targets are masked by "`LOCATION1`" and "`LOCATION2`" in the whole dataset. Following [9], we only consider the top 4 aspects (i.e. "`GENERAL`", "`PRICE`", "`TRANSIT-LOCATION`" and "`SAFETY`") when evaluate aspect detection and sentiment classification. To evaluate the robustness of our **RAEC** framework, the restaurants domain in Task 12 of Semeval 2015 (Sem15) is also utilized for model evaluation. To be comparable and consistent with SentiHood dataset, we treat the attribute (i.e. "`GENERAL`", "`PRICES`", "`QUALITY`", "`STYLE_OPTIONS`", or "`MISCELLANEOUS`") of the referred aspect category as the aspect on Sem15 dataset. The whole dataset contains 1239 sentences, 898 sentences have single target and 341 sentences have multiple targets. There are 1,197 targets in the training set and 542 targets in the testing set[4], and following [10], there is no development set. The class distribution of the two datasets is shown in **Table 1**.

TABLE 2
Example of an input sentence paired with a (target, aspect) pair and the output labels on SentiHood and Sem15 dataset.

Dataset	Sentence	Target, Aspect	Output
SentiHood	**location2** is central London so extremely expensive, **location1** is often considered the coolest area of London.	location1, GENERAL	Positive
		location1, PRICE	None
		location1, TRANSIT	None
		location1, SAFETY	None
		location2, GENERAL	None
		location2, PRICE	Negative
		location2, TRANSIT	Positive
		location2, SAFETY	None
Sem15	The **food** was very good, a great deal, and the **place** its self was great.	food, GENERAL	None
		food, PRICES	Positive
		food, QUALITY	Positive
		food, STYLE_OPTIONS	None
		food, MISCELLANEOUS	None
		place, GENERAL	Positive
		place, PRICES	None
		place, QUALITY	None
		place, STYLE_OPTIONS	None
		place, MISCELLANEOUS	None

4.2 Experimental Setting

To compare the improvement of our approach based on different pre-trained embeddings, we use GloVe [66], ELMo [18] and BERT [55] (BERT-Base, Uncased)[5] to initialize the embeddings of words. We randomly initialize $\mathbf{W}$ and $\mathbf{b}$ for all experiments. The parameter c in our experiments is set to 6, and λ is set to 0.01. Given a sentence s, a list of labels (t, a, r), corresponding to *target*, *aspect* and *sentiment*, is provided. The task of TABSA can be defined as detecting the mention of an aspect a for target t, and inferring the sentiment polarity r (i.e. *Positive*, *Negative*, or *None/Neutral*) for each (t, a) pair. **Table 2** shows an example of the input and output of the method. Following [10], [11], we use macro-average *F1*, Strict accuracy (Acc.) and AUC for the evaluation of aspect detection, and Acc. and AUC for sentiment classification. Following [10], for aspect detection, we output the label with the highest probability

4. we remove sentences containing no target and the NULL target.
5. In preliminary experiments, we tried using BERT-Large, and found that the performance is similar with BERT-Base.

for each target-aspect pair. For sentiment classification, we ignore the scores of *None*. Following [9], [11], we tackle the data unbalanced problem (*None* $\gg$ *Positive* + *Negative*) by sampling the same number of training instances within a batch randomly from each class.

4.3 Comparison Methods

We compare our **RAEC** framework with three representation refinement methods [1], [29], [72] and three pre-trained embeddings [18], [55], [66] based on 12 (T)ABSA models[6] on two datasets. The models with 'BERT+' or 'ELMo+' indicate that they use the pre-trained BERT or ELMo embeddings, and the others are based on GloVe.

- The (T)ABSA baseline models are as follow:
  LSTM-Final [9]: A bidirectional LSTM model, which takes the final states to represent the input.
  LSTM-Loc [9]: A bidirectional LSTM model, which takes the output representation at the word index corresponding to the target.
  ATAE-LSTM [43]: An attention-based bidirectional LSTM model, which incorporates the aspect information based on the attention mechanism.
  MemNet [44]: A memory-based model learns word and position attentions for a specific aspect.
  RAM [46]: A recurrent attention-based memory network which can capture sentiment features separated by a long distance.
  IAN [62]: An interactive attention network with context representation and aspect representation learned separately but interactively.
  MGAN [47]: A multi-grained attention network which can capture the word-level interactions between aspects and context.
  GCAE [63]: A gated CNN model which can effectively control the flow of sentiment according to the given aspect information.
  ASGCN-DT [60]: An aspect-specific graph neural model exploits syntactical information and word dependencies by a graph convolutional network (GCN) and the dependency tree.
  SenticLSTM [10]: A bidirectional LSTM model based on hierarchical attentions and external commonsense knowledge.
  Delayed-memory [11]: A memory-based model utilizing a delayed update mechanism to track targets for TABSA.
  BERT-pair-QA-M and **BERT-pair-NLI-M** [65]: BERT-based models, which construct an auxiliary sentence from the aspect and respectively convert the task to a sentence-pair question answering and natural language inference task.

- The pre-trained embeddings baselines are as follow:
  ELMo+models [18]: Models based on ELMo.
  BERT+models [55]: Models based on BERT.

- The comparison representation refinement methods are as follow:
  SER+models [29]: Models based on a sentiment embedding refinement method, which can capture sufficient sentiment information from sentence.
  CAE+models [1]: Models based on a context-aware embedding refinement method, which achieves the state-of-the-art performance for embedding refinement in the TABSA task.
  GBCN+models [72]: Models based on a BERT representation enhanced method, which uses a gating mechanism with context-aware aspect embeddings to enhance the BERT representation for TABSA.

- The variants of our proposed **RAEC** framework are as follow:
  Sentic+models: Models based on our framework which only use affective embedding.
  Sentic+Pos+models: Models based on our framework which only use context-affective embedding (affective embedding and position encoding), but without target and aspect refinement.
  RE+models: Models based on our framework with only target and aspect refinement and without using context-affective embeddings.
  RAEC+models: Models based on our complete embedding refinement framework.

6. Note that some models were originally designed for ABSA, but can also be applied to the TABSA task.

4.4 Main Results on Different Datasets

Table 3 shows the experimental performance on two datasets. On the SentiHood dataset, the models based on our **RAEC** framework (**RAEC**+models) achieve better performance than competitor methods for both aspect detection and sentiment classification, including the models based on embedding refinement methods (**SER**+models, **CAE**+models and **GBCN**+models) and key pre-trained methods (**ELMo**+models and **BERT**+models). This indicates that our **RAEC** framework can be directly incorporated into embedding-based TABSA models and achieve superior performance. Compared with the previous best-performing non-BERT model (Delayed-memory), our best model (**RAEC**+Delayed-memory) significantly improves aspect detection (by 3.8% in accuracy, 4.7% in F1 and 2.8% in AUC) and sentiment classification (by 4.1% in accuracy and 2.5% in AUC). For BERT-based models, our proposed **RAEC** framework also performs superiorly compared with BERT-based models (BERT-pair-QA-M and BERT-pair-NLI-M). The results indicate that our **RAEC** framework can make pre-trained embeddings more effective, with the refined embeddings for targets and aspects more discriminative in different contexts. Accordingly, the dependencies between aspects and their corresponding targets are extracted from the context to capture better sentiment features for sentiment classification of target-aspect pairs.

On the Sem15 dataset, models based on our **RAEC** framework (**RAEC**+models) also substantially outperform other models for both aspect detection and sentiment classification over both non-BERT and BERT-based models. In which, compared with the previous best-performing non-BERT model (Delayed-memory), our best model (**RAEC**+Delayed-memory) significantly improves aspect detection (by 3.6% in accuracy, 4.3% in F1 and 2.6% in AUC)

TABLE 3
Main Experimental results. Results with ♮ are retrieved from the original papers. † denotes average score over 10 runs, best scores are in bold.

Model	SentiHood					Sem15				
	Aspect Detection			Sentiment Classification		Aspect Detection			Sentiment Classification	
	Acc. (%)	F1 (%)	AUC (%)	Acc. (%)	AUC (%)	Acc. (%)	F1 (%)	AUC (%)	Acc. (%)	AUC (%)
LSTM-Final	—	68.9$^\natural$	89.8$^\natural$	82.0$^\natural$	85.4$^\natural$	64.2	68.2	83.8	71.3	74.2
ELMo+LSTM-Final†	—	71.0	91.0	83.9	86.7	65.8	71.6	84.6	73.6	76.8
BERT+LSTM-Final†	—	72.6	91.3	84.5	87.0	66.3	71.4	84.5	73.8	77.1
SER+LSTM-Final†	—	72.1	90.6	83.2	86.5	65.4	71.0	84.2	73.0	76.3
RAEC+LSTM-Final† **(ours)**	—	**76.5**	**93.4**	**87.5**	**89.6**	**70.4**	**75.3**	**87.2**	**76.0**	**80.7**
LSTM-Loc	—	69.3$^\natural$	89.7$^\natural$	81.9$^\natural$	83.9$^\natural$	63.8	68.7	82.6	72.6	74.5
ELMo+LSTM-Loc†	—	72.7	91.2	84.3	86.2	66.3	70.5	83.9	73.0	76.2
BERT+LSTM-Loc†	—	73.4	91.6	84.9	87.0	66.8	71.2	84.3	73.5	77.4
SER+LSTM-Loc†	—	71.8	89.9	82.6	86.4	65.1	69.8	83.4	72.7	75.3
RAEC+LSTM-Loc† **(ours)**	—	**76.8**	**93.7**	**87.3**	**89.2**	**69.4**	**74.5**	**86.9**	**75.8**	**80.1**
ATAE-LSTM	66.0	75.7	91.4	87.8	91.2	66.5	74.5	86.8	73.3	81.6
ELMo+ATAE-LSTM†	69.8	76.8	92.3	88.9	92.3	68.1	75.8	88.2	74.6	82.8
BERT+ATAE-LSTM†	70.1	77.6	92.7	89.3	92.7	68.9	76.6	88.7	75.3	83.1
SER+ATAE-LSTM†	68.5	76.8	91.9	89.2	92.3	67.8	75.3	87.4	75.0	82.3
RAEC+ATAE-LSTM† **(ours)**	**73.5**	**79.3**	**94.6**	**93.1**	**94.1**	**71.2**	**79.2**	**89.6**	**79.2**	**83.6**
MemNet	65.3	73.8	89.9	85.2	88.7	65.3	72.8	85.2	72.7	78.5
ELMo+MemNet†	66.4	74.9	90.7	87.3	89.1	66.5	73.1	86.2	73.2	79.2
BERT+MemNet†	67.1	76.0	91.3	88.0	89.5	66.7	72.6	85.7	73.6	80.4
SER+MemNet†	65.8	74.2	90.0	88.2	89.7	66.2	71.7	84.5	74.0	80.7
RAEC+MemNet† **(ours)**	**72.6**	**77.5**	**93.7**	**90.5**	**94.6**	**70.5**	**76.7**	**88.5**	**75.4**	**82.3**
RAM	66.7	77.8	93.0	89.2	91.9	69.4	75.8	88.3	75.0	82.4
ELMo+RAM†	68.5	78.8	93.6	90.4	92.6	69.9	76.8	88.5	75.3	82.8
BERT+RAM†	69.4	79.2	94.5	90.8	93.0	70.3	77.2	88.7	75.8	83.1
SER+RAM†	67.0	78.9	93.9	91.0	92.7	69.4	76.2	88.3	76.1	82.3
RAEC+RAM† **(ours)**	**74.1**	**81.2**	**95.3**	**94.8**	**96.2**	**72.4**	**80.7**	**91.3**	**80.7**	**84.3**
IAN	65.9	75.2	91.8	88.2	91.6	66.5	73.2	86.9	74.7	81.5
ELMo+IAN†	66.8	76.3	92.1	88.7	91.5	66.9	74.2	87.4	75.3	81.8
BERT+IAN†	67.0	76.8	92.4	89.2	92.2	67.2	74.8	87.5	75.8	82.2
SER+IAN†	66.2	76.3	91.9	90.1	92.8	66.8	73.9	87.0	76.1	82.6
RAEC+IAN† **(ours)**	**73.5**	**79.7**	**94.8**	**93.3**	**94.3**	**71.4**	**79.8**	**90.8**	**78.7**	**83.2**
MGAN	67.2	77.4	92.8	88.7	93.5	67.2	75.3	88.4	75.3	82.6
ELMo+MGAN†	68.7	78.7	93.6	90.2	94.6	68.3	76.5	89.0	76.0	83.0
BERT+MGAN†	69.6	79.3	93.8	90.7	95.2	68.9	77.1	89.6	76.3	83.7
SER+MGAN†	67.4	77.4	92.5	89.8	94.7	67.5	75.6	88.2	76.3	83.5
RAEC+MGAN† **(ours)**	**74.9**	**81.2**	**95.6**	**94.0**	**95.3**	**71.8**	**81.5**	**91.6**	**81.2**	**85.1**
GCAE	65.4	73.8	90.2	86.5	90.4	66.3	71.4	85.2	72.8	79.6
ELMo+GCAE†	66.7	74.0	90.6	87.2	90.7	66.7	72.3	85.4	72.8	79.3
BERT+GCAE†	67.0	74.5	90.8	87.2	90.9	67.1	73.0	85.7	72.6	79.8
SER+GCAE†	66.2	73.4	90.3	87.4	90.9	66.5	72.8	85.2	73.0	80.1
RAEC+GCAE† **(ours)**	**71.5**	**77.8**	**92.3**	**91.5**	**93.5**	**71.0**	**79.5**	**89.3**	**78.4**	**82.8**
ASGCN-DT	67.5	76.2	91.3	88.4	92.2	68.0	74.2	86.9	75.8	83.2
ELMo+ASGCN-DT†	68.9	77.6	92.3	89.8	93.9	69.5	76.5	89.2	77.3	83.7
BERT+ASGCN-DT†	70.2	79.8	92.8	90.5	94.6	69.7	77.2	89.5	77.6	84.1
SER+ASGCN-DT†	67.3	76.3	91.3	90.2	94.2	68.2	74.5	87.5	77.4	84.2
RAEC+ASGCN-DT† **(ours)**	**75.3**	**82.4**	**94.2**	**93.7**	**96.4**	**72.6**	**80.4**	**91.2**	**80.8**	**85.6**
SenticLSTM	67.4$^\natural$	78.2$^\natural$	91.5$^\natural$	89.3$^\natural$	92.6$^\natural$	67.3$^\natural$	76.4$^\natural$	88.3$^\natural$	76.5$^\natural$	82.1
ELMo+SenticLSTM†	70.2	79.1	92.0	91.3	93.2	68.5	77.4	89.2	77.0	82.7
BERT+SenticLSTM†	70.8	80.0	92.4	92.0	93.4	69.3	77.8	89.6	77.3	83.2
SER+SenticLSTM†	68.2	77.9	91.3	91.7	93.1	67.8	76.6	88.5	76.8	82.7
CAE+SenticLSTM†	73.8$^\natural$	79.3$^\natural$	93.8$^\natural$	93.0$^\natural$	93.7$^\natural$	71.2$^\natural$	78.6$^\natural$	90.4$^\natural$	76.8$^\natural$	83.8$^\natural$
RAEC+SenticLSTM† **(ours)**	**75.8**	**82.7**	**95.3**	**94.6**	**96.8**	**72.5**	**81.2**	**91.7**	**81.0**	**85.7**
Delayed-memory	73.5$^\natural$	78.5$^\natural$	94.4$^\natural$	91.0$^\natural$	94.8$^\natural$	70.3	77.4	90.8	76.4	83.6
ELMo+Delayed-memory†	74.2	79.7	94.7	91.7	95.2	70.8	78.7	91.2	76.5	84.0
BERT+Delayed-memory†	75.2	80.3	95.2	92.4	95.7	71.2	78.3	91.0	76.9	84.2
SER+Delayed-memory†	73.7	78.8	94.0	92.0	95.0	70.6	77.8	90.7	76.7	84.2
CAE+Delayed-memory†	76.4$^\natural$	81.0$^\natural$	96.8$^\natural$	92.8$^\natural$	96.2$^\natural$	71.6$^\natural$	79.1$^\natural$	91.8$^\natural$	77.2$^\natural$	84.6$^\natural$
RAEC+Delayed-memory† **(ours)**	**77.3**	**83.2**	**97.2**	**95.1**	**97.3**	**73.9**	**81.7**	**93.4**	**81.3**	**86.3**
BERT-pair-QA-M	79.4$^\natural$	86.4$^\natural$	97.0$^\natural$	93.6$^\natural$	96.4$^\natural$	73.8	80.1	92.9	78.7	85.2
CAE+BERT-pair-QA-M†	81.0	87.3	97.5	94.7	97.2	74.1	80.6	93.5	79.6	85.8
GBCN+BERT-pair-QA-M	81.9$^\natural$	87.6$^\natural$	97.3$^\natural$	94.5$^\natural$	97.5$^\natural$	-	-	-	-	-
RAEC+BERT-pair-QA-M† **(ours)**	**82.3**	**88.3**	**97.8**	**96.0**	**98.5**	**75.1**	**82.8**	**95.1**	**82.5**	**87.4**
BERT-pair-NLI-M	78.3$^\natural$	87.0$^\natural$	97.5$^\natural$	92.1$^\natural$	96.5$^\natural$	72.7	80.6	92.5	78.3	85.5
CAE+BERT-pair-NLI-M†	80.8	87.7	97.8	94.1	97.0	74.5	81.2	93.6	79.1	86.2
GBCN+BERT-pair-NLI-M	81.3$^\natural$	88.0$^\natural$	97.2$^\natural$	93.8$^\natural$	97.2$^\natural$	-	-	-	-	-
RAEC+BERT-pair-NLI-M† **(ours)**	**82.0**	**88.5**	**98.4**	**95.6**	**98.3**	**75.4**	**82.3**	**94.7**	**82.7**	**87.8**

TABLE 4
Ablation studies on SentiHood and Sem15.

Model	SentiHood					Sem15				
	Aspect Detection			Sentiment Classification		Aspect Detection			Sentiment Classification	
	Acc. (%)	F1 (%)	AUC (%)	Acc. (%)	AUC (%)	Acc. (%)	F1 (%)	AUC (%)	Acc. (%)	AUC (%)
SenticLSTM	67.4	78.2	91.5	89.3	92.6	67.3	76.4	88.3	76.5	82.1
Sentic+SenticLSTM†**(ours)**	67.8	78.6	93.3	89.5	93.5	67.5	76.3	88.7	76.2	81.6
Sentic+Pos+SenticLSTM† **(ours)**	72.2	79.5	94.0	93.3	94.6	70.3	78.6	89.8	77.0	83.5
RE+SenticLSTM† **(ours)**	74.9	79.8	94.2	93.0	95.0	70.6	78.5	90.4	76.6	82.9
RAEC+SenticLSTM† **(ours)**	**75.8**	**82.7**	**95.3**	**94.6**	**96.8**	**72.5**	**81.2**	**91.7**	**81.0**	**85.7**
Delayed-memory	73.5	78.5	94.4	91.0	94.8	70.3	77.4	90.8	76.4	83.6
Sentic+Delayed-memory† **(ours)**	73.8	78.7	94.8	91.2	95.3	71.1	77.9	91.0	76.6	83.8
Sentic+Pos+Delayed-memory† **(ours)**	74.8	80.2	95.3	92.8	96.2	72.4	78.7	91.6	78.1	84.9
RE+Delayed-memory† **(ours)**	75.7	81.0	96.6	92.6	96.0	72.0	79.2	92.0	77.6	84.5
RAEC+Delayed-memory† **(ours)**	**77.3**	**83.2**	**97.2**	**95.1**	**97.3**	**73.9**	**81.7**	**93.4**	**81.3**	**86.3**

TABLE 5
Experimental results of different classes.

Model	SentiHood			Sem15		
	Positive	Negative	None	Positive	Negative	None
SenticLSTM	66.2	78.1	90.3	63.1	72.5	93.5
RAEC+SenticLSTM **(ours)**	**72.9**	**84.3**	**90.7**	**70.5**	**78.9**	**93.9**
Delayed-memory	68.2	77.4	89.8	65.7	73.3	92.8
RAEC+Delayed-memory **(ours)**	**74.1**	**85.5**	**90.0**	**71.0**	**81.2**	**93.1**

and sentiment classification (by 4.9% in accuracy and 2.7% in AUC). This demonstrates the generalizability of our **RAEC** framework for embedding refinement in TABSA.

In addition, it is also worth noting that compared with the previous promising representation refinement methods (**CAE**+models and **GBCN**+models), models based on our **RAEC** framework (**RAEC**+Delayed-memory, **RAEC**+BERT-pair-QA-M and **RAEC**+BERT-pair-NLI-M) achieve better performance for aspect detection and sentiment classification on both SentiHood and Sem15. This indicates that our **RAEC** framework can help the models to better utilize and enhance both non-BERT and BERT-based representations of target and aspect through the incorporation of affective knowledge and location encoding, and improve the performance of target-aspect sentiment classification.

4.5 Ablation Study

In this section, we study the impact of different components of our **RAEC** framework based on two models (SenticLSTM and Delayed-memory). As shown in **Table 4**, when only using affective embeddings (**Sentic**+SenticLSTM and **Sentic**+Delayed-memory), the improvement of the performance over both SenticLSTM and Delayed-memory is negligible on both datasets. However, when position encoding is considered, we observe significant improvement for both aspect detection and sentiment analysis on two datasets. This shows that the relative position between the target and each affective word is important in the TABSA task. We also note that the models without contextual affective embeddings (**RE**+SenticLSTM and **RE**+Delayed-memory) only marginally improve the performance compared with previous methods, and are much worse than models based on our complete **RAEC** framework (**RAEC**+Delayed-memory and **RAEC**+SenticLSTM) for sentiment classification in particular. This indicates that incorporating context-affective embeddings is essential for target-aspect sentiment analysis.

4.6 Analysis of Class Distribution

As shown in Table 1, the class distribution of the datasets is imbalanced. Hence, in this section, we analyze the performance of target-aspect sentiment classification, i.e. detecting the sentiment polarity of a sentence according to a given target-aspect pair, in different classes on the two benchmark datasets and report the Macro F1-scores over different classes in **Table 5**. We can observe that the performance of the models fluctuates considerably in different classes, which potentially indicates that the performance of the model in different classes is influenced by the size of the data. It is also worth noting that our proposed framework (**RAEC**) achieves outstanding improvement in all classes and reveals smaller gaps in different classes. This implies that models based on our proposed **RAEC** can effectively balance the learning between different classes in the case of imbalanced data distribution.

4.7 Effect of Word Embeddings

The above experiments reveal that the models based on our embedding refinement framework (**RAEC**) achieve the best performance for both aspect detection and sentiment analysis. In this section, we study the impact of word embeddings on our **RAEC** framework by reporting the accuracy of using different embeddings and show the results in **Figure 3**. We can see that embeddings from randomly initialized, GloVe, ELMo and BERT can be directly incorporated into our **RAEC** framework and achieve better performance than the original models. We also notice that models based on pre-trained embeddings (e.g. GloVe, ELMo and BERT) achieve

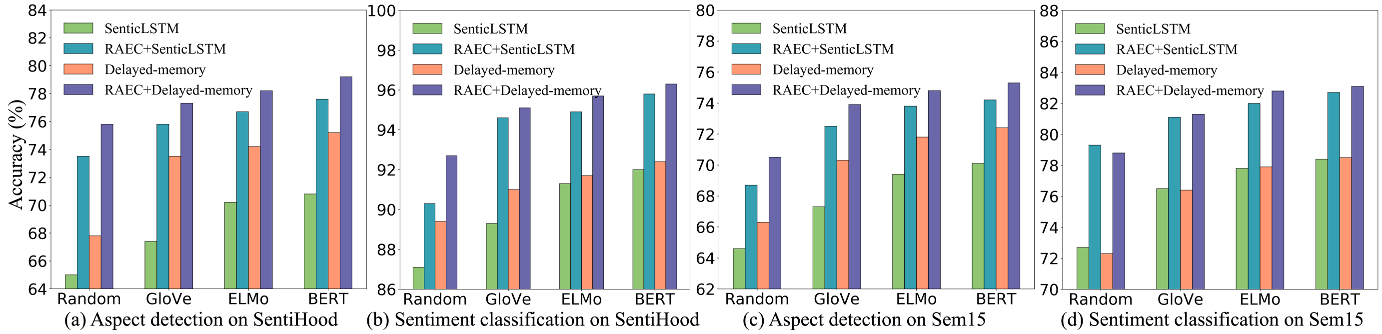


Fig. 3. Experimental results of using different pre-trained word embeddings and randomly initialized word embeddings.

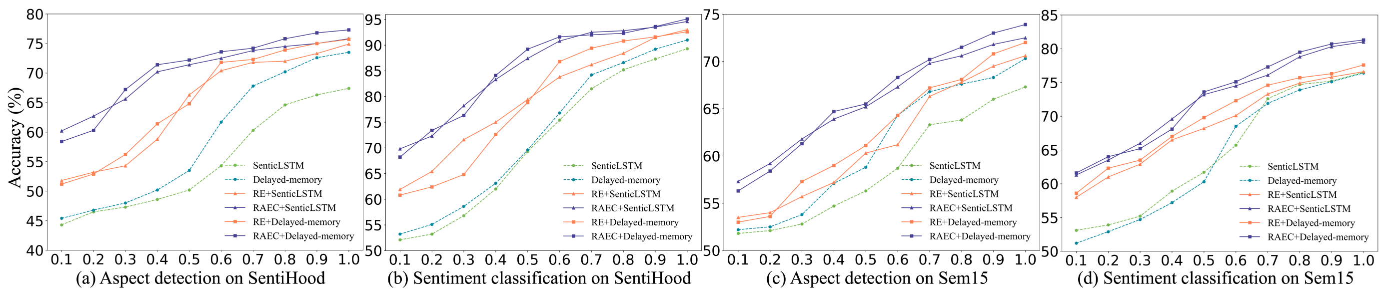


Fig. 4. Experimental results of using different ratios of training data. The ratios of training data are varied from 0.1 to 1.

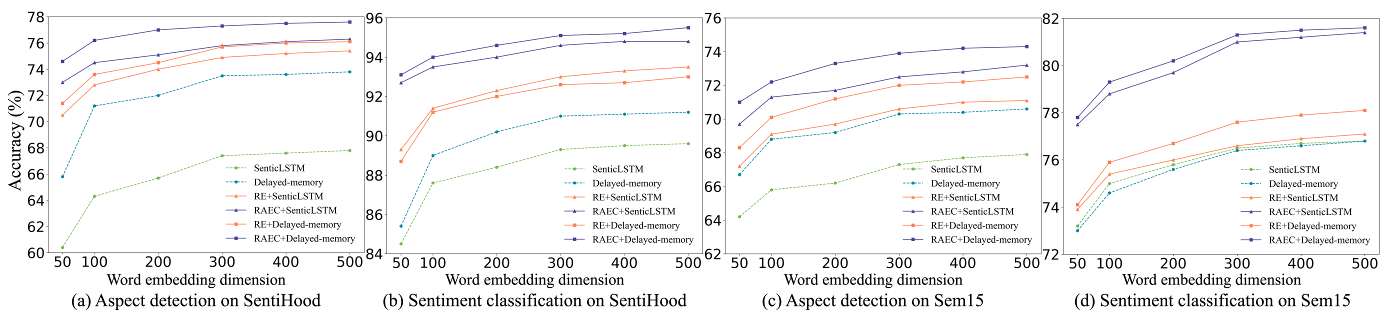


Fig. 5. Experimental results of using different dimensions of word embeddings.

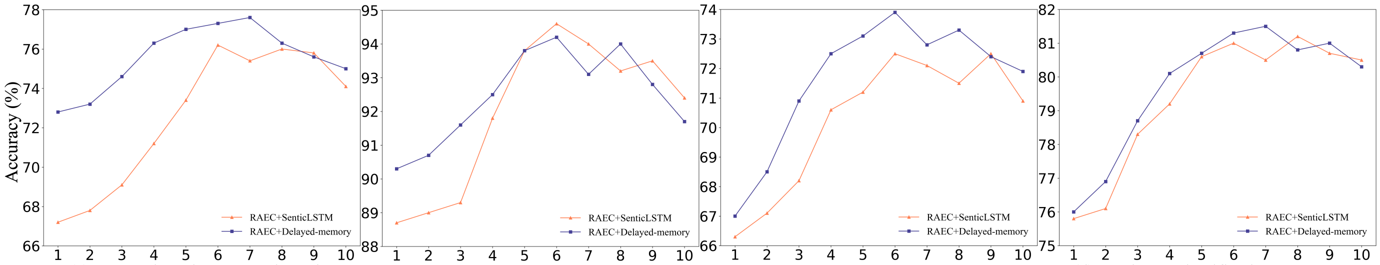
significant improvement in comparison with randomly initialized embeddings. This shows that models using pre-trained embeddings can extract better sentiment features for targets and aspects from the context.

4.8 Effect of Training Data Size

To evaluate the impact of the size of training data on our **RAEC** framework, we conduct experiments using different ratios of training data of the two datasets based on Glove embeddings, and show the results in **Figure 4**. We can observe that the models based on our **RAEC** framework beat the competitor models consistently with different training data sizes for both aspect detection and sentiment classification, and the improvement is more prominent with only using 10%-60% training data. In **Figure 4(a)** and **4(c)**, we compare the models with and without contextual affective embeddings on aspect detection. It is observed that the models based on contextual affective embedding (with **RAEC**) achieve more significant improvements when using a small set of training data. In addition, as shown in **Figure 4(b)** and **4(d)**, significant improvement is also observed by **RAEC**+models with a small set of training data for sentiment classification in comparison with **RE**+models. The results show that the **RAEC**+models are more robust to the varying sizes of training data.

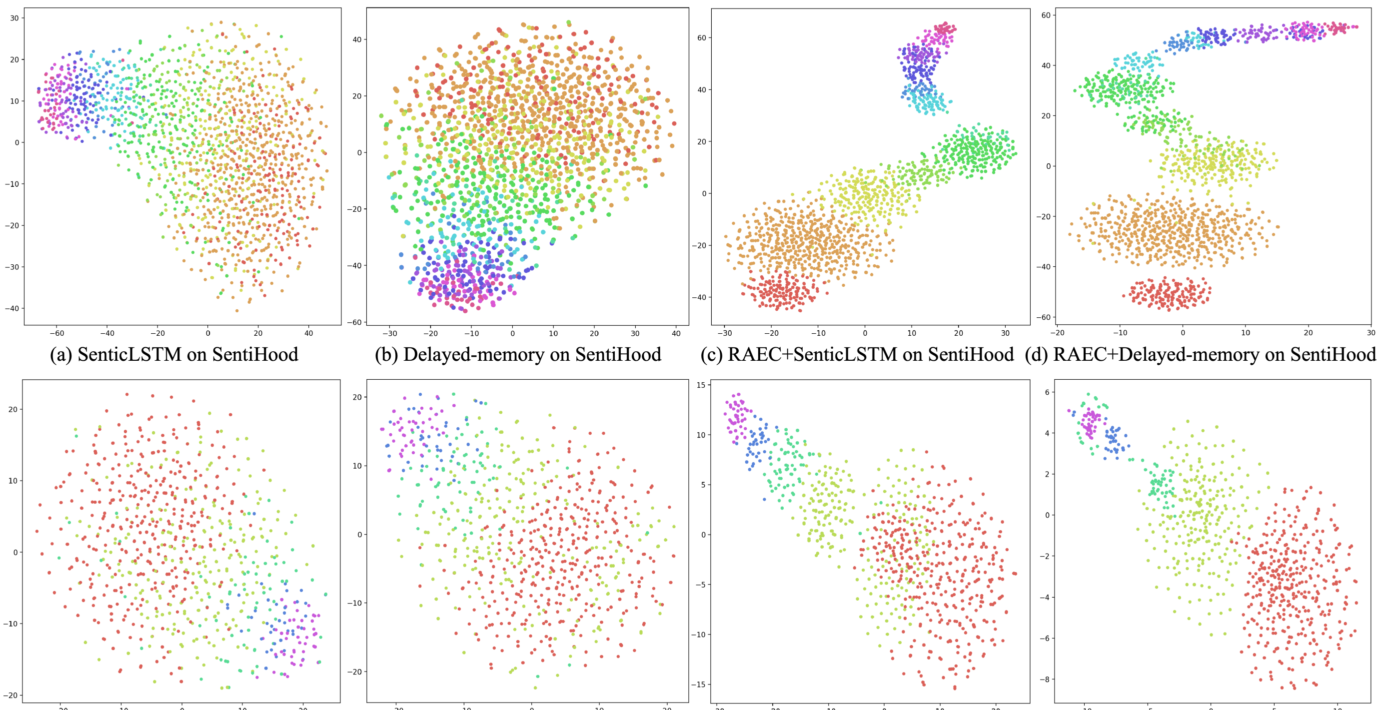
4.9 Effect of Dimensions of Word Embeddings

To analyze the effect of using different dimensions of word embeddings for both aspect detection and sentiment analysis, we conduct experiments on the two datasets with SenticLSTM and Delayed-memory, and show the results in **Figure 5**. We train original word embeddings with different dimensions on the Wikipedia corpus based on GloVe, and compute their corresponding affective embeddings and encode position vectors with corresponding dimension sizes. Compared with the original models (SenticLSTM and Delayed-memory) and the models only using target and aspect refinement (**RE**+SenticLSTM and **RE**+Delayed-memory), the models based on our complete framework (**RAEC**+SenticLSTM and **RAEC**+Delayed-memory) achieve the best performance with different word embedding dimensions for both aspect detection and sentiment analysis.



(a) Aspect detection on SentiHood (b) Sentiment classification on SentiHood (c) Aspect detection on Sem15 (d) Sentiment classification on Sem15

Fig. 6. Experimental results of using different values of c. The values of c are varied from 1 to 10.



(a) SenticLSTM on SentiHood (b) Delayed-memory on SentiHood (c) RAEC+SenticLSTM on SentiHood (d) RAEC+Delayed-memory on SentiHood

(e) SenticLSTM on Sem15 (f) Delayed-memory on Sem15 (g) RAEC+SenticLSTM on Sem15 (h) RAEC+Delayed-memory on Sem15

Fig. 7. The visualization of learning intermediate aspect vectors. Different colors represent different aspects.

We also observe that with the increasing number of word embedding dimensions, the accuracy of both aspect detection and sentiment classification increases for all the models. And it saturates when the word embedding dimension goes beyond 300. It is also noteworthy that while the performance of all the other models drops significantly when the word embedding dimension is 50, the results of the models based on our **RAEC** framework only degrade slightly, which showing the effectiveness of our proposed embedding refinement approach. To balance the performance and computational cost, we suggest to set the dimension to 300.

4.10 Effect of the value of c

In this section, we explore the effect of the value of c in our experiments for both aspect detection and sentiment analysis on the two datasets with **RAEC**+SenticLSTM and **RAEC**+Delayed-memory. The results are shown in **Figure 6**. We can observe that the accuracy of both aspect detection and sentiment classification increases with the increasing value of c and peaks when $c = 6$. Further increasing the values of c results in the worse performance for both aspect detection and sentiment analysis. Intuitively, the small value of c leads to the limited affective information to be used for embedding refinement. However, the excessive value of c will introduce spurious affective words into the embedding refinement process of targets and aspects. Hence in our experiments, the optimal value of c is 6.

4.11 Visualization of Intermediate Aspect Vectors

To qualitatively demonstrate how the **RAEC** framework improves the performance of TABSA, we employ all 12 predefined aspects from SentiHood and 5 predefined aspects from Sem15 to visualize the intermediate vectors of contextual affective aspect embeddings $\tilde{\mathbf{a}}$ and the original aspect embeddings $\mathbf{a}$ learned by SenticLSTM and Delayed-memory models. Here, we ignore the label of "*None*". We extract the intermediate aspect embeddings from both models, and apply t-SNE [73] to project the high-dimensional representations to the two-dimensional space. As demonstrated in **Figure 7**, each point in the figure represents an

Id	Sentence	Target	Aspect	Label
1	I would suggest LOCATION1 as a pretty good area	LOCATION1	general	positive
2	LOCATION1 is a bit run down , I would n't recommend it	LOCATION1	general	negative
3	LOCATION1 is a good place to live with a major rail station	LOCATION1	transit	positive
	LOCATION1 is a good place to live with a major rail station	LOCATION1	live	positive
4	If I had the money, I would live near LOCATION1	LOCATION1	live	positive
	If I had the money, I would live near LOCATION1	LOCATION1	price	negative
5	I also like LOCATION1 and LOCATION2 for going out	LOCATION1	nightlife	positive
	I also like LOCATION1 and LOCATION2 for going out	LOCATION2	nightlife	positive
6	I got my flat broken into LOCATION1 but never LOCATION2	LOCATION1	safety	negative
	I got my flat broken into LOCATION1 but never LOCATION2	LOCATION2	safety	positive
7	LOCATION1 is a bad hole , LOCATION2 is ok but out of the way	LOCATION1	general	negative
	LOCATION1 is a bad hole , LOCATION2 is ok but out of the way	LOCATION2	transit	negative
8	Avoid LOCATION1 , near LOCATION2 is very nice to live	LOCATION1	general	negative
	Avoid LOCATION1 , near LOCATION2 is very nice to live	LOCATION2	live	positive

Fig. 8. Examples of refining specific aspects by extracting highly relevant words from the context. The color depth represents the value of the weight, the darker the large value. Words without color background represent values are 0. "transit" represents the aspect "`TRANSIT-LOCATION`".

aspect embedding and different colors indicate the different type of aspects. Compared with original aspect embeddings, our contextual affective aspect embeddings display a clearer separation of different aspects. It shows that the refined aspect embeddings, generated via their corresponding targets and associated affective words in context, exhibit a greater discrimination among aspects, which is beneficial to model learning for both aspect detection and sentiment classification.

4.12 Case Study

In this section, we present case studies with some typical examples selected from all 12 predefined aspects on SentiHood dataset to better analyze how the proposed **RAEC** framework works in extracting highly associated words from input sentences for refining aspect representations with the help of the sparse coefficient vector. The results are shown in **Figure 8**. To make it easier to visualize the results, we highlight the words with different color intensities showing their corresponding weights of the sparse coefficient vectors obtained by the proposed **RAEC** framework. **Example 1** and **Example 2** are two sentences with the same target-aspect pair but opposing polarities. Our proposed embedding refinement framework is able to incorporate the contextually-relevant affective words for refining target and aspect embeddings. Since those words bear different polarities, the resulting target and aspect embeddings in these two sentences would be different and eventually contribute to the correct sentiment classification results for target-aspect pairs. **Example 3** is the sentence containing two different aspects but with the same sentiment polarity. We can see that the most relevant contextual words extracted for these two aspects are the same. But the refined aspect representations are different as evidenced by the different coefficient weights. The coefficient weight of the word *live* for the aspect "`live`" is evidently higher than that of the same word for the aspect "`transit`". **Example 4** is the sentence containing two different aspects with opposing sentiment polarities. Higher coefficient weights were correctly placed on the aspect-indicative words when classifying the sentiment polarity for different target-aspect pairs. **Example 5** to **Example 8** are sentences with multiple (two) different targets, which are more challenging for both aspect detection and sentiment analysis in the task of TABSA. **Example 5** contains the same aspect and the same sentiment polarity but for different targets. We observe higher coefficient weights on the desired targets when considering different target-based sentiment classification. **Example 6** expresses opposing polarities on different target-aspect pairs. We observe that different affective words are extracted to refine the target and aspect embeddings for different target-aspect pairs by the proposed **RAEC** framework. **Example 7** contains two different target-aspect pairs with the same sentiment polarity. Overlapping affective words are extracted to refine different aspect representations. Nevertheless, the higher coefficient weights are placed on the correct target of interest when considering different target-aspect pairs. **Example 8** is the sentence contains different target-aspect pairs with different sentiment polarities. For this kind of sentence, our **RAEC** framework can extract different affect-indicative words for different target-aspect pairs to refine the representations of aspects by the coefficient vectors. This shows that affective words close to a specific target can be extracted by the target representation refinement component, and such information can be used to refine the corresponding aspect representation, which eventually leads to more accurate sentiment classification results of the target-aspect pairs.

TABLE 6
Examples of error analysis.

	Example	Label
1	If you don't mind living quite a long way out places like location1 and location2 are also cheap and quite nice	(location1, price, Positive) (location1, general, Positive) (location1, transit, Negative) (location2, price, Positive) (location2, general, Positive) (location2, transit, Negative)
2	location1 is really nice - I live in location2	(location1, general, Positive)
3	There are better places than location1 to go hang out	(location1, general, Negative)

4.13 Error Analysis

We conduct error analysis on the experimental results. We found that most of the errors can be broadly represented by three types of examples, as shown in **Table 6**. **Example 1** is a very complex sentence with intricate sentiment relations between targets and aspects. Hence models find it difficult to detect all the aspects for a given target and infer the sentiment for all target-aspect pairs correctly. It would be interesting to see if incorporating syntactic features could assist models to learn latent relations between targets and aspects better and thus lead to the improvement of the performance. Another type of errors occurred when there are multiple targets in a sentence but some of them has no associated sentiment, as shown in **Example 2**. Models are more inclined to classify the sentiment towards "`location2`" as *positive* due to the occurrence of the phrase "*really nice*". The final type of errors is shown in **Example 3** which contains a phrase expressing an indirect sentiment towards the target "`location1`". Existing models have a difficulty in dealing with such cases.

5 Conclusion

In this paper, we present a novel embedding refinement framework for targeted aspect-based sentiment analysis (TABSA), namely **RAEC**. To generate more informative embeddings for the task of TABSA, commonsense knowledge and relative location information are incorporated into the refinement of contextual word embeddings. Afterwards, a set of highly relevant words are selected from the context to refine the embeddings of targets and aspects. Such embedding refinement makes the model detect aspects more accurately for a given target and also achieve better accuracy on classifying the sentiment polarities of target-aspect pairs. As a universal embedding refinement framework, the proposed **RAEC** framework can be easily integrated with existing embedding-based TABSA models. Experimental results demonstrated the effectiveness and robustness of the **RAEC** framework for the task of TABSA. Future work may include incorporating syntactic dependencies between targets and aspects in the context to obtain higher quality refined embeddings for both targets and aspects.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (61632011, 61876053, 62006062), the Guangdong Province Covid-19 Pandemic Control Research Funding (2020KZDZX1224), the Shenzhen Foundational Research Funding (JCYJ20180507183527919 and JCYJ20200109113441941), China Postdoctoral Science Foundation (2020M670912), Joint Lab of Lab of HITSZ and China Merchants Securities, the UK EPSRC (grant no. EP/T017112/1, EP/V048597/1), Youth Innovation Promotion Association of CAS China (No. 2020357), and Shenzhen Science and Technology Innovation Program (Grant No. KQTD20190929172835662). Yulan He is supported by a Turing AI Fellowship funded by the UK Research and Innovation (UKRI) (grant no. EP/V020579/1).

References

[1] B. Liang, J. Du, R. Xu, B. Li, and H. Huang, "Context-aware embedding for targeted aspect-based sentiment analysis," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Jul. 2019, pp. 4678–4683.
[2] B. Pang, L. Lee *et al.*, "Opinion mining and sentiment analysis," *Foundations and Trends® in Information Retrieval*, vol. 2, no. 1–2, pp. 1–135, 2008.
[3] B. Liu, "Sentiment analysis and opinion mining," *Synthesis lectures on human language technologies*, vol. 5, no. 1, pp. 1–167, 2012.
[4] J. Lipenkova, "A system for fine-grained aspect-based sentiment analysis of chinese," in *Proceedings of ACL-IJCNLP 2015 System Demonstrations*. Association for Computational Linguistics and The Asian Federation of Natural Language Processing, 2015, pp. 55–60.
[5] T. H. Nguyen and K. Shirai, "Phrasernn: Phrase recursive neural network for aspect-based sentiment analysis," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, September 2015, pp. 2509–2514.
[6] W. Wang, J. P. Sinno, and D. Daniel, "Memory networks for fine-grained opinion mining," *Artificial Intelligence*, vol. 265, pp. 1–17, 2018.
[7] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. AL-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, V. Hoste, M. Apidianaki, X. Tannier, N. Loukachevitch, E. Kotelnikov, N. Bel, S. M. Jiménez-Zafra, and G. Eryiğit, "Semeval-2016 task 5: Aspect based sentiment analysis," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 2016, pp. 19–30.
[8] D. Tang, B. Qin, X. Feng, and T. Liu, "Effective LSTMs for target-dependent sentiment classification," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 3298–3307.
[9] M. Saeidi, G. Bouchard, M. Liakata, and S. Riedel, "Sentihood: Targeted aspect based sentiment analysis dataset for urban neighbourhoods," in *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016*, 2016, pp. 1546–1556.
[10] Y. Ma, H. Peng, and E. Cambria, "Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM," in *Proceedings of the Thirty-Second Conference on Artificial Intelligence,(AAAI-18)*, 2018, pp. 5876–5883.
[11] F. Liu, T. Cohn, and T. Baldwin, "Recurrent entity networks with delayed memory update for targeted aspect-based sentiment analysis," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018, pp. 278–283.
[12] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of machine learning research*, vol. 3, no. Feb, pp. 1137–1155, 2003.
[13] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: a simple and general method for semi-supervised learning," in *Proceedings of the 48th annual meeting of the association for computational linguistics*. Association for Computational Linguistics, 2010, pp. 384–394.
[14] T. Mikolov, G. Corrado, K. Chen, and J. Dean, "Efficient estimation of word representations in vector space," in *Proceedings of ICLR*, 01 2013, pp. 1–12.

[15] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[16] O. Levy and Y. Goldberg, "Neural word embedding as implicit matrix factorization," in *Advances in neural information processing systems*, 2014, pp. 2177–2185.

[17] M. Li, Q. Lu, Y. Long, and L. Gui, "Inferring affective meanings of words from word embedding," *IEEE Transactions on Affective Computing*, vol. 8, no. 4, pp. 443–456, 2017.

[18] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 2227–2237.

[19] M. E. Peters, M. Neumann, R. Logan, R. Schwartz, V. Joshi, S. Singh, and N. A. Smith, "Knowledge enhanced contextual word representations," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, 2019, pp. 43–54.

[20] K. Ethayarajh, "How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 55–65.

[21] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2014, pp. 1746–1751.

[22] J. Wang, L.-C. Yu, K. R. Lai, and X. Zhang, "Dimensional sentiment analysis using a regional cnn-lstm model," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2016, pp. 225–230.

[23] M. Dragoni, S. Poria, and E. Cambria, "Ontosenticnet: A commonsense ontology for sentiment analysis," *IEEE Intelligent Systems*, vol. 33, no. 3, pp. 77–85, 2018.

[24] Z. Jianqiang, G. Xiaolin, and Z. Xuejun, "Deep convolution neural networks for twitter sentiment analysis," *IEEE Access*, vol. 6, pp. 23 253–23 260, 2018.

[25] B. Shi, Z. Fu, L. Bing, and W. Lam, "Learning domain-sensitive and sentiment-aware word embeddings," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 2494–2504.

[26] J. Du, L. Gui, Y. He, R. Xu, and X. Wang, "Convolution-based neural attention with applications to sentiment classification," *IEEE Access*, 2019.

[27] J. Wang, L.-C. Yu, K. R. Lai, and X. Zhang, "Investigating dynamic routing in tree-structured LSTM for sentiment analysis," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3432–3437.

[28] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin, "Learning sentiment-specific word embedding for twitter sentiment classification," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2014, pp. 1555–1565.

[29] L.-C. Yu, J. Wang, K. R. Lai, and X. Zhang, "Refining word embeddings for sentiment analysis," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Sep. 2017, pp. 534–539.

[30] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[31] L.-C. Yu, J. Wang, K. R. Lai, and X. Zhang, "Refining word embeddings using intensity scores for sentiment analysis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 3, pp. 671–681, 2017.

[32] A. Utsumi, "Refining pretrained word embeddings using layer-wise relevance propagation," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 4840–4846.

[33] C. Yonghe, H. Lin, L. Yang, Y. Diao, S. Zhang, and F. Xiaochao, "Refining word reesprentations by manifold learning," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. AAAI Press, 2019, pp. 5394–5400.

[34] L. Wang, Z. Cao, G. de Melo, and Z. Liu, "Relation classification via multi-level attention CNNs," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 1298–1307.

[35] Y. Hao, Y. Zhang, K. Liu, S. He, Z. Liu, H. Wu, and J. Zhao, "An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 221–231.

[36] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[37] S. Narayan, R. Cardenas, N. Papasarantopoulos, S. B. Cohen, M. Lapata, J. Yu, and Y. Chang, "Document modeling with external attention for sentence extraction," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 2020–2030.

[38] Y. Yang, D. Zhou, Y. He, and M. Zhang, "Interpretable relevant emotion ranking with event-driven attention," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 177–187.

[39] H. Dong, W. Wang, K. Huang, and F. Coenen, "Joint multi-label attention networks for social text annotation," 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)*, 2019, pp. 1348–1354.

[40] Y. Huang and J. Du, "Self-attention enhanced CNNs and collaborative curriculum learning for distantly supervised relation extraction," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 389–398.

[41] P. Li, K. Mao, X. Yang, and Q. Li, "Improving relation extraction with knowledge-attention," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 229–239.

[42] Y. Yang, Z. Deyu, Y. He, and M. Zhang, "Interpretable relevant emotion ranking with event-driven attention," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 177–187.

[43] Y. Wang, M. Huang, x. zhu, and L. Zhao, "Attention-based lstm for aspect-level sentiment classification," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 606–615.

[44] D. Tang, B. Qin, and T. Liu, "Aspect level sentiment classification with deep memory network," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 214–224.

[45] C. Yang, H. Zhang, B. Jiang, and K. Li, "Aspect-based sentiment analysis with alternating coattention networks," *Information Processing & Management*, vol. 56, no. 3, pp. 463–478, 2019.

[46] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent attention network on memory for aspect sentiment analysis," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 452–461.

[47] F. Fan, Y. Feng, and D. Zhao, "Multi-grained attention network for aspect-level sentiment classification," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 3433–3442.

[48] M. Hu, S. Zhao, L. Zhang, K. Cai, Z. Su, R. Cheng, and X. Shen, "CAN: Constrained attention networks for multi-aspect sentiment analysis," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 4601–4610.

[49] X. Tan, Y. Cai, and C. Zhu, "Recognizing conflict opinions in aspect-level sentiment classification with dual attention networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3426–3431.

[50] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent twitter sentiment

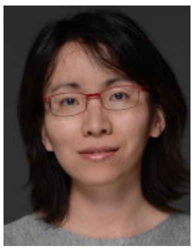
classification," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 49–54.
[51] D.-T. Vo and Y. Zhang, "Target-dependent twitter sentiment classification with rich automatic features," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.
[52] M. Yang, Q. Qu, X. Chen, C. Guo, Y. Shen, and K. Lei, "Feature-enhanced attention network for target-dependent sentiment classification," *Neurocomputing*, vol. 307, pp. 91–97, 2018.
[53] X. Li, L. Bing, W. Lam, and B. Shi, "Transformation networks for target-oriented sentiment classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2018, pp. 946–956.
[54] Z. Gao, A. Feng, X. Song, and X. Wu, "Target-dependent sentiment classification with bert," *IEEE Access*, vol. 7, pp. 154 290–154 299, 2019.
[55] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 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)*, 2019, pp. 4171–4186.
[56] W. Wang, S. J. Pan, D. Dahlmeier, and X. Xiao, "Recursive neural conditional random fields for aspect-based sentiment analysis," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, November 2016, pp. 616–626.
[57] S. Ruder, P. Ghaffari, and J. G. Breslin, "A hierarchical model of reviews for aspect-based sentiment analysis," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 999–1005.
[58] J. Tang, Z. Lu, J. Su, Y. Ge, L. Song, L. Sun, and J. Luo, "Progressive self-supervised attention learning for aspect-level sentiment analysis," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 557–566.
[59] Z. Lei, Y. Yang, M. Yang, W. Zhao, J. Guo, and Y. Liu, "A human-like semantic cognition network for aspect-level sentiment classification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 6650–6657.
[60] C. Zhang, Q. Li, and D. Song, "Aspect-based sentiment classification with aspect-specific graph convolutional networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, 2019, pp. 4567–4577.
[61] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, "An interactive multi-task learning network for end-to-end aspect-based sentiment analysis," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 504–515.
[62] D. Ma, S. Li, X. Zhang, and H. Wang, "Interactive attention networks for aspect-level sentiment classification," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, 2017, pp. 4068–4074.
[63] W. Xue and T. Li, "Aspect based sentiment analysis with gated convolutional networks," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018, pp. 2514–2523.
[64] Q. Jiang, L. Chen, R. Xu, X. Ao, and M. Yang, "A challenge dataset and effective models for aspect-based sentiment analysis," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 6280–6285.
[65] C. Sun, L. Huang, and X. Qiu, "Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence," 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)*, 2019, pp. 380–385.
[66] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543.
[67] D. Tang, F. Wei, B. Qin, N. Yang, T. Liu, and M. Zhou, "Sentiment embeddings with applications to sentiment analysis," *IEEE transactions on knowledge and data Engineering*, vol. 28, no. 2, pp. 496–509, 2015.
[68] A. Weichselbraun, S. Gindl, F. Fischer, S. Vakulenko, and A. Scharl, "Aspect-based extraction and analysis of affective knowledge from social media streams," *IEEE Intelligent Systems*, vol. 32, no. 3, pp. 80–88, 2017.
[69] E. Cambria, J. Fu, F. Bisio, and S. Poria, "Affectivespace 2: Enabling affective intuition for concept-level sentiment analysis," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
[70] E. Cambria, S. Poria, R. Bajpai, and B. Schuller, "Senticnet 4: A semantic resource for sentiment analysis based on conceptual primitives," in *Proceedings of COLING 2016, the 26th international conference on computational linguistics: Technical papers*, 2016, pp. 2666–2677.
[71] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos, "Semeval-2015 task 12: Aspect based sentiment analysis," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 2015, pp. 486–495.
[72] X. Li, X. Fu, G. Xu, Y. Yang, J. Wang, L. Jin, Q. Liu, and T. Xiang, "Enhancing bert representation with context-aware embedding for aspect-based sentiment analysis," *IEEE Access*, vol. 8, pp. 46 868–46 876, 2020.
[73] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

Bin Liang is a Ph.D. Student at the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. His current research interests include natural language processing, sentiment analysis, deep learning and machine learning.

Rongdi Yin is a master student at the School of Computer Science and Technology, Harbin Institution of Technology, Shenzhen, China. His current research interests include natural language processing, text classification and sentiment analysis.

Jiachen Du received the Ph.D. degree from Harbin Institute of Technology, Shenzhen, China. His research interests include sentiment analysis, dialogue generation, and deep learning.

Lin Gui is a Research Fellow at the Department of Computer Science in the University of Warwick. He received Ph.D. degree in Computer Science from Harbin Institute of Technology, China, specializing in Machine Learning, text mining, natural language processing. His research interests include text classification algorithms, natural language understanding, sentiment analysis, stance detection, emotion cause detection, and topic modeling.



Yulan He is a Professor of Computer Science in the University of Warwick, U.K. She is a Turing AI Fellow. She has published over 170 papers in the areas of natural language understanding, sentiment analysis and opinion mining, question-answering, topic/event extraction from text, biomedical text mining, and social media analytics. Yulan received her Ph.D. degree in spoken language understanding from the University of Cambridge, U.K.

Min Yang is currently an Associate Professor at Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. She received the Ph.D. degree from the University of Hong Kong. Her current research interests include natural language processing, deep learning and machine learning.

Ruifeng Xu received the Ph.D. degree in computer science from The Hong Kong Polytechnic University. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. He has published more than 150 papers in natural language processing, sentiment analysis, and social media analysis.