TARGET-AWARE MEMORY NETWORKS FOR ASPECT SENTIMENT CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Memory networks with external memories and the attention mechanism have been used in numerous applications and achieved superior results. Many recent studies have also made great efforts to capture more accurate attentions. In this paper, we focus on solving an important sentiment analysis problem, called aspect sentiment classification (ASC), which requires us to extend the traditional memory networks because in ASC, the classification can be dependent on the target aspect. For example, given two target aspects (which are product features) resolution and price, “higher resolution” and “lower price” express positive opinions while “higher price” and “lower resolution” express negative opinions. We observe that the same word higher (or lower) can indicate different sentiments for different target aspects. Traditional memory networks have difficulty in capturing such target-aware (dependent) sentiment signals. In this paper, we propose the target-aware memory networks to tackle the problem, which is able to model sentiment interaction between target aspects and the context words jointly with the attention mechanism. Our experimental results demonstrate its effectiveness.

1 INTRODUCTION

Memory networks with external memories and the attention mechanism have been applied to many natural language processing (NLP) tasks such as question answering and machine translation. In this work, we focus on solving a particular sentiment analysis problem, called aspect sentiment classification (ASC) (Liu [2012]), using memory networks. Although memory networks have been applied to solve this problem and have given promising results (Tang et al. [2016]). However, we found that using traditional memory networks to solve the ASC problem has a major shortcoming. That is, they are not able to model the interaction between aspects and contextual sentiment signals/indicators, which is important in many cases.

Before discussing the above issue, let us first give the task definition of ASC: given a target aspect and a sentence containing the aspect, we want to classify the sentiment/opinion polarity (positive, negative, or neutral) expressed in the sentence about the aspect. The aspects are also referred to as opinion targets (or targets), which are usually product features/attributes in product reviews. Note that in practice, the aspects can be specified by the user or extracted automatically using an existing aspect extraction technique (Liu [2012]). In this work, we assume that the aspects are given.

We use some sample opinion/sentiment sentences to illustrate the ASC task and also the need to model sentiment and aspect interactions. For example, we have four sentences: (1) “the screen resolution is excellent,” (2) “the screen resolution is excellent but the price is ridiculous,” and (3) “the price is very high,” and (4) “the screen resolution is very high.” In sentence (1), the sentiment expressed on the aspect picture is positive. In sentence (2), the sentiment expressed on the aspect screen resolution (or resolution for short) is positive but the sentiment expressed on the aspect price is negative. We note that before classifying the sentiment polarity, we first need to identify the sentiment context about the given aspect. Memory networks (MN) are capable of performing this task because when a specified target is given, its important sentiment context can be captured by the attention mechanism, e.g., “excellent” will be identified as an important sentiment context for resolution and “ridiculous” will be pointed out for price. Prior studies (Tang et al. [2016]) have shown the effectiveness of MNs in tackling this problem. However, if we look at sentences (3) and (4), we realize that the above mechanism is not sufficient. In these two sentences, the sentiment
contexts are both “high”, but sentence (3) is negative and (4) is positive simply because the target aspects are different. Note that the aspects themselves indicate no sentiment. Thus, in ASC, we need target-aware (or -dependent) sentiment, meaning that the sentiment of a context word is conditioned on the assigned target.

In spite of their superior performance, existing MNs have difficulties to deal with the target-aware sentiment issue. Their capability of attention alignment is definitely useful, because it can point out the important context words for a given aspect, and in many cases the sentiment can be directly inferred from the attended context words. From the above sentence (1), we can see that if the main attention is placed correctly on “excellent”, which is an aspect-independent sentiment word, classifying the sentiment expressed as positive is fairly easy. Attentions can help classifying sentence (2) as well. However, for the sentences (3) and (4), placing the correct attention on “high” is necessary but not sufficient, as its sentiment is conditioned on the aspect. We will gain more theoretical insight of this problem with MNs in Section 3. To tackle this problem, we propose target-aware memory networks, which can model target-context sentiment interactions jointly with the attention mechanism in their memories. Our experimental results will demonstrate the effectiveness of our proposed techniques.

2 MEMORY NETWORK FOR ASC

In this section we describe our basic memory network for ASC. It does not include the proposed aspect-context sentiment interaction techniques, which will be introduced in Section 4. The design is similar to previous works (Sukkbaatar et al., 2015; Tang et al., 2016) except we use a different attention alignment function (shown in Eq. 1). Their models will be compared in our experiments as well.

Input Representation: Given a target aspect \( t \in \mathbb{R}^V \), an embedding matrix \( A \in \mathbb{R}^{d \times V} \) is used to convert it to a vector representation, \( v_t = At \). Similarly, each context word (each of the other non-aspect words in the sentence) \( x_i \in \{x_1, x_2, \ldots, x_n\} \) is also projected to the continuous space stored in memory, denoted as \( m_i = Ax_i \in \{m_1, m_2, \ldots, m_n\} \). Here \( n \) is the number of words in a sentence and \( i \) indicates the word in position \( i \).

Attention: Attention can be obtained based on the above input representations. Specifically, an attention score \( \alpha_i \) for the context word \( x_i \) is computed based on the alignment function in Eq. 1, where \( M \in \mathbb{R}^{d \times d} \) is a general learning matrix suggested by Luong et al. (2015).

\[
\alpha_i = \text{softmax}(v_t^T M m_i)
\] (1)

In this manner, attention \( \alpha = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \) is represented as a vector of probabilities, indicating the weight/importance of different context words towards a given target aspect.

Output Representation: Another embedding matrix \( C \) is used for each context word \( x_i \) for generating its individual continuous vector \( c_i = C x_i \). A response/output vector \( o \) is produced by summing over the transformed vectors weighted by the probability vectors \( \alpha \).

\[
o = \sum_i \alpha_i c_i
\] (2)

Sentiment Scores: The aspect sentiment scores for classes (positive, neutral, or negative) are then calculated. A sentiment-specific weight matrix \( W \in \mathbb{R}^{K \times d} \) is used. The sentiment scores \( s \) are represented in a vector \( s \in \mathbb{R}^K \), where \( K \) is the number of (sentiment) classes, which is 3 in ASC.

\[
s = W(o + u)
\] (3)

Final Prediction: The final sentiment probability \( y \) is produced with a softmax operation. The class with the highest probability is used as the categorical prediction \( \hat{y} = \max_k(y_k) \) where \( k \) is the index of a sentiment class.

\[
y = \text{softmax}(s)
\] (4)
3 Problem with the above Model for ASC

As indicated in the introduction, in this section we analyze the target-aware sentiment issue in the above model in detail. We first expand the sentiment score calculation from Eq. 3 to its individual terms (Eq. 5). We further transform it to a linear combination of $Wc_i$ and $Wv_t$ as shown in Eq. 6.

$$s = W(o + v_t) = W(\sum_i \alpha_i c_i + v_t)$$  \hspace{1cm} (5)

$$s = W(\sum_i \alpha_i c_i + v_i) = \alpha_1 Wc_1 + \alpha_2 Wc_2 + ... \alpha_i Wc_i + Wv_t$$  \hspace{1cm} (6)

A problem with Eq. 6 for ASC is that the sentiment effect of a context word $\alpha_i Wc_i$ does not explicitly depend on the target word $t$. When the final sentiment scores from the context and the target are summed together, they can be viewed as contributing separately to achieve the final score, which does not reflect their dependency relationship that we identified earlier. As discussed in introduction, a context word can express a totally different sentiment on a different target (e.g., “high”). Ideally, the sentiment score for each context word should not be independently calculated without considering the target. Therefore, Eq. 6 has a shortcoming in dealing with the target-aware sentiment.

We use our previous examples to gain further insight. Let us say we have two (target) aspect words price and resolution (denoted as $p$ and $r$). We also have two possible context words “high” and “low” (denoted as $h$ and $l$). As these two sentiment words can modify both two aspects, we can construct four snippets “high price”, “low price”, “high resolution” and “low resolution”, which we can regard as four short review sentences. Their sentiments are negative, positive, positive, negative respectively. Let us set $W$ to be $\mathbb{R}^{1 \times d}$ so $s$ becomes a 1-dimensional sentiment score indicator. $s > 0$ indicates a positive sentiment and $s < 0$ indicates a negative sentiment. Based on the above examples we have four corresponding inequalities: (a) $W(\alpha_h v_h + v_p) < 0$, (b) $W(\alpha_l v_l + v_p) > 0$, (c) $W(\alpha_h v_h + v_r) > 0$ and (d) $W(\alpha_l v_l + v_r) < 0$. We can drop all $\alpha$ terms here as they all equal to 1, i.e., they are the only context word in the snippets to attend to (the target words are not contexts). From (a) and (b) we have (e) $Wv_h < -Wv_p < Wv_l$. From (c) and (d) we have (f) $Wv_l < -Wv_r < Wv_h$. From (e) and (f) we have (g) $Wv_h < Wv_l < Wv_h$, which is a contradiction.

Clearly, real-world review sentences are usually longer than these short expressions, but since the attention mechanism will focus on sentiment signals (e.g., “high” and “low”), we face the same problem. Furthermore, non-sentiment context words should carry no sentiment information. For example, in a sentence like “the price is high”, context words “the” and “is” will not help address the above problem.

One may then ask whether the attention mechanism can help address the problem, i.e., the $\alpha$ terms cannot be dropped in a long sentence so they can affect the final results too. However, while the adjustment of attention can affect the final sentiment score, it can only assign higher or lower weight to increase or reduce the sentiment score of a context word. It cannot change the intrinsic sentiment polarity, which is determined by $Wc_i$ only. We will show examples in our experiments.

4 The Proposed Approaches

This section presents our proposed approaches, target-aware memory networks, which include five alternative techniques. This section also discusses related implementation details.

Non-Linear (NL): This is the first approach which utilizes a non-linear projection to couple context and aspect. Instead of directly using the linear combination to calculate the sentiment score as shown in Eq. 5, we use a non-linear projection ($\tanh$) as replacement, shown in Eq. 7.

$$s = W \cdot \tanh(o + v_t) = W \cdot \tanh(\sum_i \alpha_i c_i + v_t)$$  \hspace{1cm} (7)

$$W \cdot \tanh(\sum_i \alpha_i c_i + v_t) = W \cdot \tanh(\sum_i \alpha_i (c_i + v_t)) = W \cdot \tanh(\sum_i \alpha_i (c_i'))$$  \hspace{1cm} (8)

In this technique, the target vector $v_t$ will affect the context by shifting their representation to $c_i'$ (shown in Eq. 8). In this manner, the vector of the same context word will be changed differently.
However, for explanation purposes we still use and found that using the dot product the context word “good” can be used as a strong and aspect independent sentiment indicator. (2)

prediction needs target-aware interaction. In a simple sentence like “the screen resolution is good”,

Under review as a conference paper at ICLR 2018

Interaction Term (IT): The second proposed approach is to add explicit aspect-context iteration items. Specifically, Eq[5] is replaced by Eq[9]. Four new factors are involved, namely, \( d_t, d_f, W_s, w_f \) \((d_t, d_f \in \mathbb{R}^{d \times 1}, W_s \in \mathbb{R}^{K \times d} \) and \( w_f \in \mathbb{R}^{d} \)). \( W_s \) models the sentiment effect for \( c_t, d_t \) and \( d_f \) are new vector representations for context \( i \) and aspect \( t \) produced from a new embedding matrix \( D \), i.e., \( d_t = Dx_i, d_f = Dt_t \). Different from the aforementioned embeddings, \( D \) is specifically designed for capturing the sentiment interaction effect between context and aspect words, as they will affect the final sentiment score directly via \( w_f < d_t, d_f > \). \( w_f \) is a sentiment-specific vector and \( < d_t, d_f > \) denotes the dot product of the two vectors \( d_t \) and \( d_f \). Compared to the basic memory network, this model can capture aspect-aware sentiment signals because the interactions between a context word \( w \) and different aspect words (say, \( p \) and \( r \)) can be different, i.e., \( < d_w, d_p > \neq < d_w, d_r > \).

Coupled Interaction (CI): This proposed technique couples aspect-context interaction with a context output representation, defined by Eq[10]. Specifically, \( e_i \) is another output representation for \( x_i \), which is coupled with aspect-context interaction factor \( < d_t, d_f > \). For each context word \( x_i, e_i \) is generated as \( e_i = Ex_i \) where \( E \) is an embedding matrix. \( < d_t, d_f > \) and \( e_i \) function together as a target-aware contextual vector and are used to produce sentiment scores with \( W_f(W_f \in \mathbb{R}^{K \times d}) \).

\[
\begin{align*}
    s &= W_s \alpha + w_f \sum_i a_i < d_t, d_f > = \sum_i a_i(W_s c_i + w_f < d_t, d_f >) \\
    &= W_s \alpha + W_f \sum_i a_i < d_t, d_f > e_i = \sum_i a_i(W_s c_i + W_f < d_t, d_f > e_i) \\
    &= W_s \alpha + W_f \sum_i a_i < d_t, d_f > c_i = \sum_i a_i(W_s c_i + W_f < d_t, d_f > c_i)
\end{align*}
\]

Joint Coupled Interaction (JCI): A variant of the above model is to replace \( e_i \) with \( c_i \), which means to learn a joint output representation (Eq[11]). This can reduce learning parameters and simplify the CI model.

\[
\begin{align*}
    s &= W_s \alpha + W_f \sum_i a_i < d_t, d_f > c_i = \sum_i a_i(W_s c_i + W_f < d_t, d_f > c_i)
\end{align*}
\]

Joint Projected Interaction (JPI): This model also employs a unified output representation like JCI, but a context output vector \( c_i \) will be projected to two different continuous spaces before sentiment score calculation. To achieve the goal, two projection matrices \( W_1, W_2 \) and the non-linear projection function \( \tanh \) are used. The intuition is that, when we want to reduce the (embedding) parameters and learn a joint representation, two different sentiment effects may need to be considered in different vector spaces. The two sentiment effects are modeled as two terms shown in Eq[13] the first term can be viewed as a target-independent sentiment effect of a context word while the second term captures the target-aware sentiment interaction. A joint sentiment-specific weight matrix \( W_j(W_j \in \mathbb{R}^{K \times d}) \) is used to control and balance the interplay between these two effects.

\[
\begin{align*}
    s &= W_j \sum_i a_i \tanh(W_1 c_i) + \sum_i a_i < d_t, d_f > \tanh(W_2 c_i) \\
    &= \sum_i a_i W_j \tanh(W_1 c_i) + \sum_i a_i W_j < d_t, d_f > \tanh(W_2 c_i)
\end{align*}
\]

Notes. (1) In the above models their first terms are still needed, because not in all cases a sentiment prediction needs target-aware interaction. In a simple sentence like “the screen resolution is good”, the context word “good” can be used as a strong and aspect independent sentiment indicator. (2) The aspect-context interaction can be calculated by other functions. We have tried several methods and found that using the dot product \( < d_t, d_f > \) or \( d_t W d_f \) (with a projection matrix \( W \)) can produce good results. In fact, \( d_t W d_f \) generates slightly better results and we adopt it in our experiments. However, for explanation purposes we still use \( < d_t, d_f > \) consistently in this paper.
Loss function: The proposed models are all trained in an end-to-end manner by minimizing the cross entropy loss. Let us denote a sentence and a target aspect as $x$ and $t$ respectively. They appear together in a pair format as input and all such pairs construct the dataset $D$. $g^k_{x,t}$ denotes a gold sentiment label of a $(x,t)$ pair for sentiment class $k$. $y_{x,t}$ is a model-predicted sentiment distribution based on the $(x,t)$ pair, and $y^k_{x,t}$ denotes its probability for class $k$. Based on them, the training loss can be constructed as Eq. 14.

$$
\text{loss} = - \sum_{(x,t) \in D} \sum_{k \in K} g^k_{x,t} \log y^k_{x,t}
$$

5 RELATED WORK

Studies about memory networks, attention mechanisms and aspect sentiment classification (ASC) are related to our work. Memory networks (Weston et al., 2014; Sukhbaatar et al., 2015), equipped with external storage and the attention mechanism, have become state-of-the-art neural models. The large explicit storage is usually endowed with continuous representation (Graves et al., 2014), which can be effectively used for attention calculation. The attention mechanism has also been widely used in various tasks such as image recognition (Mnih et al., 2014), machine translation (Bahdanau et al., 2014), reading comprehension (Hermann et al., 2015) and sentence summarization (Rush et al., 2015). (Cho et al., 2015) provided a comprehensive review and (Luong et al., 2015) introduced different attention alignment functions. Meanwhile, another track of research lies in learning better attentions, by considering more advanced attention architectures. For example, Kumar et al. (2016) designed a dynamic and recurrent attention method and Kim et al. (2017) proposed to learn structured attention by incorporating graphical models. Different from them, the focus of this work is not on improving attention learning (it does not solve our problem, discussed in section 3), but to make further use of the memory storage for jointly learning target-aware sentiment effects.

Sentiment analysis (Liu, 2012) is an important NLP problem and ASC is a fundamental task of sentiment analysis. Different from document level or sentence level sentiment classification (Pang et al., 2002; Socher et al., 2011; Kim, 2014; Yang et al., 2016), ASC is a fine-grained analysis that identifies sentiment polarities based on given target aspects. The above studies cannot be directly applied to the ASC task as they are not capable of mining finer-grained aspect dependent sentiments. The closest related work to ours comes from Yang et al. (2016), where the authors applied a traditional memory network variant to the ASC task. However, the target-aware sentiment effect is not considered, i.e., the target-aware sentiment issue remains in their model. To the best of our knowledge, our work is the first to exploit the external memory to jointly capture target-context sentiment interactions, in addition to attention learning.

6 EXPERIMENTS

We perform experiments on two popular review datasets of domains Laptop and Restaurant from SemEval 2014 (Pontiki et al., 2014). In these datasets, aspect-based sentiment polarities are manually labeled as negative, neutral and positive. The training and test sets have also been provided. Each record in each dataset is a \{sentence, aspect, sentiment\} triple, i.e., a review sentence, an aspect/target and its corresponding aspect sentiment polarity. Our task is to classify the sentiment polarity on an specified (target) aspect in a review sentence. Full statistics of the datasets are shown in Table 1.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Train</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Negative</td>
<td>Neutral</td>
<td>Positive</td>
<td>Negative</td>
<td>Neutral</td>
<td>Positive</td>
</tr>
<tr>
<td>Restaurant</td>
<td>807</td>
<td>637</td>
<td>2164</td>
<td>196</td>
<td>196</td>
<td>728</td>
</tr>
<tr>
<td>Laptop</td>
<td>870</td>
<td>464</td>
<td>994</td>
<td>128</td>
<td>169</td>
<td>341</td>
</tr>
</tbody>
</table>

Table 1: Statistics of the datasets.

6.1 CANDIDATE MODELS FOR COMPARISON

Baseline 1 (BL1): The classic end-to-end memory network (Sukhbaatar et al., 2015).
Baseline 2 (BL2): A state-of-the-art and representative memory network used for aspect sentiment
Table 2: Results on dataset Restaurant. Top three F-Macro scores are marked in bold.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>BL1</td>
<td>58.91</td>
<td>38.17</td>
<td>32.41</td>
<td>80.46</td>
<td>62.86</td>
<td>60.35</td>
<td>44.37</td>
<td>83.11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BL2</td>
<td>63.82</td>
<td>43.56</td>
<td>36.15</td>
<td>85.19</td>
<td>66.46</td>
<td>65.57</td>
<td>46.64</td>
<td>87.16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BL3</td>
<td>64.34</td>
<td>45.86</td>
<td>35.19</td>
<td>85.78</td>
<td>65.71</td>
<td>63.83</td>
<td>46.91</td>
<td>86.39</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NL</td>
<td>64.62</td>
<td>42.21</td>
<td>38.78</td>
<td>85.62</td>
<td>65.98</td>
<td>64.18</td>
<td>47.86</td>
<td>85.90</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IT</td>
<td>58.37</td>
<td>44.44</td>
<td>38.46</td>
<td>85.78</td>
<td>65.98</td>
<td>64.18</td>
<td>47.86</td>
<td>85.90</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CI</td>
<td>66.78</td>
<td>48.32</td>
<td>38.51</td>
<td>86.51</td>
<td>68.64</td>
<td>67.11</td>
<td>51.47</td>
<td>87.33</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JCI</td>
<td>66.21</td>
<td>46.23</td>
<td>38.51</td>
<td>86.65</td>
<td>68.64</td>
<td>67.11</td>
<td>51.47</td>
<td>87.33</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JPI</td>
<td>66.58</td>
<td>47.60</td>
<td>38.71</td>
<td>86.71</td>
<td>68.76</td>
<td>67.62</td>
<td>49.63</td>
<td>87.24</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Results on dataset Laptop. Top three F-Macro scores are marked in bold.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>BL1</td>
<td>56.16</td>
<td>47.06</td>
<td>45.81</td>
<td>75.63</td>
<td>60.61</td>
<td>55.59</td>
<td>45.94</td>
<td>80.29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BL2</td>
<td>60.01</td>
<td>52.67</td>
<td>47.89</td>
<td>79.48</td>
<td>65.16</td>
<td>60.00</td>
<td>52.56</td>
<td>82.91</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BL3</td>
<td>62.89</td>
<td>57.16</td>
<td>49.51</td>
<td>81.99</td>
<td>67.11</td>
<td>63.10</td>
<td>54.53</td>
<td>83.69</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NL</td>
<td>62.63</td>
<td>56.43</td>
<td>49.62</td>
<td>81.83</td>
<td>67.79</td>
<td>63.17</td>
<td>56.27</td>
<td>83.92</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IT</td>
<td>63.07</td>
<td>57.01</td>
<td>50.62</td>
<td>81.58</td>
<td>66.23</td>
<td>61.43</td>
<td>53.69</td>
<td>83.57</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CI</td>
<td>63.65</td>
<td>57.33</td>
<td>52.60</td>
<td>81.02</td>
<td>66.79</td>
<td>61.80</td>
<td>55.30</td>
<td>83.26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JCI</td>
<td>63.09</td>
<td>57.80</td>
<td>50.71</td>
<td>80.76</td>
<td>67.23</td>
<td>61.08</td>
<td>57.49</td>
<td>83.11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JPI</td>
<td>64.53</td>
<td>58.62</td>
<td>51.71</td>
<td>83.25</td>
<td>65.16</td>
<td>59.01</td>
<td>54.25</td>
<td>82.20</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

classification [Tang et al., 2016]. The main difference from BL1 is its attention alignment function, which concatenates the distributed representations of the context and aspect, and uses an additional weight matrix for attention calculation, following the method introduced in (Bahdanau et al., 2014).

Baseline 3 (BL3): This is our proposed basic memory network, which does not have the target-dependent interaction component. This baseline also allows us to show the effect of modeling target-aware sentiment signal by comparing it with the target-aware models below.

Target-Aware models (TA): The 5 proposed approaches Non-Linear (NL), Interaction Terms (IT), Coupled Interaction (CI), Joint Coupled Interaction (JCI) and Joint Projected Interaction (JPI) are used to set 5 different target-aware memory networks.

6.2 Training and Settings

We use the open-domain word embeddings[1] trained on a large Google News corpus for the initialization of all word vectors. We randomize other model parameters from a uniform distribution U(-0.05, 0.05). The dimension of the word embedding and the size of the hidden layers are 300. The learning rate is set to 0.01 and the dropout rate is set to 0.9 for all models. Stochastic gradient descent is used as our optimizer to minimize the cross-entropy error of sentiment classification. We follow other training settings from the study of end-to-end memory network (Sukhbaatar et al., 2015). The position encoding is used, following (Sukhbaatar et al., 2015) and Tang et al. (2016). We also compared the candidate models in their multiple-hops version. The number of hops is set to 3 as used in the above previous works.

6.3 Evaluation Measure

Since we have a three-class classification task and the classes are imbalanced as shown in Table 1 (i.e., the positive samples are much more populous than the negative and neutral samples), we use F1-score as our evaluation measure. We report both F1-Macro (averaged F1-score) over all classes and all class-based F1 scores so as to gain a comprehensive insight.

6.3.1 Result Analysis

The classification results of Restaurant and Laptop are shown in Table 4 and Table 3. The left part and the right part in each table show the F1 scores for the 1-hop version and 3-hops version of all

---

<table>
<thead>
<tr>
<th>Record 1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
</tr>
<tr>
<td><strong>Target</strong></td>
</tr>
<tr>
<td><strong>Goal</strong></td>
</tr>
<tr>
<td><strong>TA model</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>NTA model</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Analysis</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Record 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
</tr>
<tr>
<td><strong>Target</strong></td>
</tr>
<tr>
<td><strong>Goal</strong></td>
</tr>
<tr>
<td><strong>TA model</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>NTA model</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Analysis</strong></td>
</tr>
</tbody>
</table>

Table 4: Sample Records and Model Comparison between TA and NTA.

models respectively. The top three F-Macro scores are marked in bold in each setting. Based on them, we have the following observations:

1. In the 1-hop setting, we can see significant performance gains achieved by the target-aware models on both Restaurant and Laptop, which are shown by the overall Macro-F1 scores. We should note that the negative (Neg.) and neural (Neu.) classes as minority classes are usually harder to predict due to the lack of training information. However, we can see the target-aware models have better performance consistently due to the target-dependent information modeling. For example, we can see that JCI improves BL3 by 3.78% in the F1 of negative class (Neg.) on Restaurant. Overall, JPI performs the best in this setting.

2. In the 3-hops setting, the target-aware models achieve much better results on Restaurant. JCI, IT and CI achieve the best scores, outperforming the best baseline model BL2 by 2.38%, 2.18% and 2.03% respectively. On Laptop, BL3 and other target-aware models except JPI perform similarly. However, BL3 performs much poorer on Restaurant, only better than BL1.

3. In both the 1-hop and 3-hops settings, the highest scores are always achieved by target-aware models, which implies the usefulness of capturing target-dependent signals. While comparing the target-aware models, we found that JCI is most stable, as it always obtains the top-three scores in both 2 datasets and 2 settings. It may indicate that the joint and coupled output embeddings can help better parameter fitting between different layers/hops.

4. It is important to note that these improvements are quite large considering that in many cases target-aware interactions are not necessary because the sentiment words themselves can already tell the sentiment polarity without knowing the target, e.g., “great battery life”, and “good resolution”.

6.4 Effect of Target-Context Sentiment Interaction: Case Studies

Here we present some real sample result records to demonstrate the effectiveness of modeling target-context interaction. First, we present two sentences that require target-context interaction for sentiment classification. We then focus on comparing the performance of a target-aware (TA) model and a non-target-aware (NTA) model. Specifically, the JPI and BL3 are used. Other TA/NTA models have similar performance qualitatively to JPI/BL3 in terms of the effect of interaction modeling so
we do not list their comparisons. These two representative models are selected because they generally achieve the best classification results in TA and NTA groups in the 1-hop setting. We show the prediction results of their 1-hop version as in this setting the target-context interaction is less complicated and more linearly trackable/separable, meaning that the sentiment contribution (or sentiment score) of a single context word indexed by $i$ can be calculated by $\alpha_i W_s c_i$ (from Eq.6) and $\alpha_i W_j \tanh(W_1 c_i) + \alpha_i W_j < d_i, d_j > \tanh(W_2 c_i)$ (from [12]) respectively, each of which is a 3-dimensional vector denoting the sentiment scores for negative, neutral, and positive.

Table 3 shows two real records in Laptop. In record 1, the row of Data means that when price is targeted in the sentence “Price was higher when purchased on MAC” the expressed sentiment is negative. The row of Goal indicates that to correctly identify its sentiment polarity, the target-aware interaction between context word “higher” and price in this sentence/snippet is the key. We then report the specific sentiment scores of the word “higher” towards negative, neutral and positive classes in both TA and NTA models in the rows of TA Model and NTA Model. We can see that both of them correctly assign highest sentiment scores to the negative class, as presented in the row of Analysis. We can also observe that in the NTA model the first value (3.3641) of the 3-dimension vector $\{3.3641, -0.3275, -0.0750\}$ calculated by $\alpha_i W_s c_i$ is greater than the second ($-0.3275$) and the third values ($-0.0750$). Notice that $\alpha_i$ is always positive (as attention scores range in $0, 1$ with softmax), so it can be inferred that the first value in vector $W_s c_i$ is greater than the other two values. We will refer to this inference again below.

Now we move forward to record 2 in Table 3. In this record the target is Resolution and its sentiment is positive in the sentence “(MacBook) Air has higher resolution.” Similarly, in this case we need the correct interaction between aspect resolution and context word “higher”. Notice that although we have the same context word “higher”, different from record 1, this interaction needs to be positive (see Goal). In rows of TA Model and NTA Model, we find that while the TA model assigns the highest sentiment score of word “higher” to positive class correctly, the NTA model gives it to the negative class wrongly. This error is understandable if we refer to the above inference, i.e., we already know that in this NTA model the first value in vector $W_s c_i$ is greater than the other two values when $i$ is the word “higher”. The reason of this unavoidable mistake is that $W_s c_i$ is not coupled with the target term. In contrast, $W_j < d_i, d_j > \tanh(W_2 c_i)$ can adjust the sentiment polarity with the aspect vector $d_i$ encoded. Other TA models like JCI can also realize it by $W_j < d_i, d_j > c_i$.

One may say that the aspect information ($v_t$) is actually also considered in the form of $\alpha_i W_s c_i + W_s v_t$ in NTA models which should make a difference, i.e., $W_s v_t$ may help address the problem given different $v_t$. Yes, it may help, which means in the above example a NTA model can make $W_s v_{resolution}$ favor the positive class and $W_s v_{price}$ favor the negative class. But then we will have trouble when the context word is “lower”, where it requires $W_s v_{resolution}$ to favor negative and $W_s v_{price}$ to favor positive in order to help. We can see the contradictions. They reflect the theoretical problem discussed in the above section 3.

In addition to this case study, we also found many other interesting target-aware sentiment like “large bill” and “large portion”, “small tip” and “small portion” in Restaurant. Furthermore, it is worth noting that the TA model improves the identification of neutral sentiments as well. For instance, it generates the sentiment score vector for the context word “over” for target aspect price as $\{0.1373, 0.0066, -0.1433\}$ (negative) and for target aspect dinner as $\{0.0496, 0.0591, -0.1128\}$ (neutral), but the NTA model produces both negative scores $\{0.0069, 0.0025, -0.0090\}$ and $\{0.0078, 0.0028, -0.0102\}$ for the two different target aspects.

7 Conclusion and Future Work

In this paper, we first introduced the aspect-dependent sentiment issue in ASC and indicated the difficulty of modeling it using traditional memory networks. After that, we discussed the basic memory network for ASC and analyzed the reason why it is not capable of capturing aspect-dependent sentiment from a theoretical perspective. We then presented our five alternative approaches for constructing target-aware memory networks. Finally, we reported the experimental results quantitatively and presented some case studies, which demonstrate the effectiveness of our proposed approaches. One direction of future work is to improve the attention learning and target-aware interaction at the same time. We can also try to integrate with other more advanced attention architectures and dependency relationships.
REFERENCES


