Incorporating Dynamic Semantics into Pre-Trained Language Model for Aspect-based Sentiment Analysis

Kai Zhang\textsuperscript{1}, Kun Zhang\textsuperscript{2}, Mengdi Zhang\textsuperscript{3}, Hongke Zhao\textsuperscript{4}, Qi Liu\textsuperscript{1,*}, Wei Wu\textsuperscript{3}, Enhong Chen\textsuperscript{1}

\textsuperscript{1} School of Data Science, University of Science and Technology of China
\textsuperscript{2} School of Computer Science and Information Engineering, Hefei University of Technology
\textsuperscript{3} Meituan; \textsuperscript{4} College of Management and Economics, Tianjin University

kkzhang0808@mail.ustc.edu.cn; \{qiliuqi, cheneh\}@ustc.edu.cn
\{zhangkun, wuwei19850318, mdzhang\}@gmail.com; hongke@tju.edu.cn

Abstract

Aspect-based sentiment analysis (ABSA) predicts sentiment polarity towards a specific aspect in the given sentence. While pre-trained language models such as BERT have achieved great success, incorporating dynamic semantic changes into ABSA remains challenging. To this end, in this paper, we propose to address this problem by Dynamic Re-weighting BERT (DR-BERT), a novel method designed to learn dynamic aspect-oriented semantics for ABSA. Specifically, we first take the Stack-BERT layers as a primary encoder to grasp the overall semantic of the sentence and then fine-tune it by incorporating a lightweight Dynamic Re-weighting Adapter (DRA). Note that the DRA can pay close attention to a small region of the sentences at each step and re-weigh the vitally important words for better aspect-aware sentiment understanding. Finally, experimental results on three benchmark datasets demonstrate the effectiveness and the rationality of our proposed model and provide good interpretable insights for future semantic modeling.

1 Introduction

Aspect-based sentiment analysis is a branch of sentiment analysis, which aims to identify sentiment polarity of the specific aspect in a sentence (Jiang et al., 2011). For example, given a sentence “The restaurant has attentive service, but the food is terrible.”, the task aims to predict the sentiment polarities towards “service” and “food”, which should be positive and negative respectively.

As a fundamental technology, the ABSA task has broad applications, such as recommender systems (Chin et al., 2018; Zhang et al., 2021b) and question answering (Wang et al., 2019). Therefore, a great amount of research has been attracted from both academia and industry. Among them, deep neural networks (DNN) (Nguyen and Shirai, 2015; Tang et al., 2015, 2016; Zheng et al., 2020), attention mechanism (Wang et al., 2016; Ma et al., 2017) and graph neural/attention networks (Huang and Carley, 2019; Zhang et al., 2019a; Wang et al., 2020) have significantly improved the performance through deep feature alignment between the aspect representations and context representations.

Recently, the large-scaled pre-trained language models, such as Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019), realize a breakthrough for improving many language tasks, which further attracts considerable attention to enhance the semantic representations. In ABSA, Xu et al. (2019a) designed BERT-PT, which explores a novel post-training approach on the BERT model. Song et al. (2019) further proposed a text pair classification model BERT-SPC, which prepares the input sequence by appending the aspects into the contextual sentence. Although great success has been achieved by the above studies, some critical problems remain when directly applying attention mechanisms or fine-tuning the pre-trained BERT in the task of ABSA.

Specifically, most of the existing approaches select all the important words from a contextual sentence at one time. However, according to neuroscience studies, the essential words during semantic comprehension are dynamically changing with the reading process and should be repeatedly considered (Kuperberg, 2007; Tononi, 2008; Brouwer et al., 2021). For example, when judging the sentiment polarity of the aspect “system memory” in a review sentence “It could be a perfect laptop if it would have faster system memory and its radeon would have DDR5 instead of DDR3”, the important words should change from general sentiment words {“faster”, “perfect”, “laptop”} to aspect-aware words {“would have”, “faster”, “could”, “be”, “perfect”}. Through these dynamic changes, the sentiment polarity will change from positive to the ground truth sentiment label negative.

* Corresponding author.
Meanwhile, simply initializing the encoder with a pre-trained BERT does not effectively boost the performance in ABSA as we expected (Huang and Carley, 2019; Xu et al., 2019a; Wang et al., 2020). One possible reason could be that training on two specific tasks, i.e., Next Sentence Prediction and Masked LM, with rich resources leads to better semantic of the overall sentences. However, the ABSA task is conditional, which means the model needs to understand the regional semantics of sentences by fully considering the given aspect. For instance, BERT tends to understand the global sentiment of the above sentence “It could be a perfect laptop ... of DDR3” regardless of which aspect is given. But in ABSA, the sentence is more likely to be different sentiment meanings for different aspects (e.g., negative for “system memory” while positive for “DDR5”). Therefore, the vanilla BERT is hardly to pay closer attention to relevant information for the specific aspect, especially when there are multiple aspects in one sentence.

To equip the pre-trained models with the ability to capture the aspect-aware dynamic semantics, we present a Dynamic Re-weighting BERT (DR-BERT) model, which considers the aspect-aware dynamic semantics in a pre-trained learning framework. Specifically, we first take the Stack-BERT layers as primary sentence encoder to learn overall semantics of the whole sentences. Then, we devise a Dynamic Re-weighting Adapter (DRA), which aims to pay most careful attention to a small region of the contextual sentence and dynamically select and re-weight one critical word at each step for better aspect-aware sentiment understanding. Finally, to overcome the limitation of vanilla BERT mentioned above, we incorporate the light-weighted DRA into each BERT encoder layer and fine-tune it to adapt to the ABSA task. We conduct extensive experiments on three widely-used datasets where the results demonstrate the effectiveness, rationality and interpretability of the proposed model.

2 Related Work

2.1 Aspect-based Sentiment Analysis

Aspect-based sentiment analysis identifies specific aspect’s sentiment polarity in the sentence. Some approaches (Ding and Liu, 2007; Jiang et al., 2011; Kiritchenko et al., 2014) designed numerous rules-based models for ABSA. For example, Ding and Liu (2007) first performed dependency parsing to determine sentiment polarity about the aspects.

In recent years, most research studies make use of the attention mechanism to learn the word’s semantic relation (Tang et al., 2015, 2016; Wang et al., 2016; Ma et al., 2017; Xing et al., 2019; Liang et al., 2019; Zhang et al., 2021a). Among them, Wang et al. (2016) proposed an attention-based LSTM to identify important information relating to the aspect. Ma et al. (2017) developed an interactive attention to model the aspect and sentence interactively. Fan et al. (2018) defined a multi-grained network to link the words from aspect and sentence. Li et al. (2018) designed a target-specific network to integrate aspect information into sentence. Tan et al. (2019) introduced a dual attention to distinguish conflicting opinions.

In addition, another research trend is to leverage syntactic knowledge to learn syntax-aware features of the aspect (Tang et al., 2019; Huang and Carley, 2019; Zhang et al., 2019a; Sun et al., 2019; Wang et al., 2020; Tang et al., 2020; Chen et al., 2020; Li et al., 2021; Tian et al., 2021). For example, Tang et al. (2020) developed dependency graph enhanced dual-transformer network to fuse the flat representations. More recently, pre-trained methods have been proved remarkably successful in the ABSA task. Song et al. (2019) devised an attentional encoder and a BERT-SPC model to learn features between aspect and context. Wang et al. (2020) reshaped the dependency trees and proposed a relational graph attention network to encode the syntax relation feature. Tian et al. (2021) explicitly utilize dependency types with a type-aware graph networks to learn aspect-aware relations.

However, these methods largely ignore the procedure of dynamic semantic comprehension (Kuperberg, 2007; Kuperberg and Jaeger, 2016; Wang et al., 2017; Zheng et al., 2019; Zhang et al., 2019c; Brouwer et al., 2021) and can not fully reveal dynamic semantic changes of the aspect-related words. Thus, it’s hard for ABSA models to achieve the same performance as human-level sentiment understanding.

2.2 Human Semantic Comprehension

Actually, no matter in the early days or now, imitating the procedure of human semantic comprehension has always been one of the original intention of many studies (Bezdek, 1992; Wang et al., 2017; Zheng et al., 2019; Li et al., 2019; Zhang et al., 2019d; Peng et al., 2020; Golan et al., 2020), such as machine reading comprehension (Zhang et al., 2019d; Peng et al., 2020), visual object detecting (Spampinato et al., 2017) and relevance estima-
Figure 1: An illustration of the proposed framework. The blue blocks constitute a pre-trained BERT model which are frozen during fine-tuning, and the right block represents the dynamic re-weighting adapter that is inserted after each BERT encoder layer and trained during fine-tuning. Moreover, $S$ and $A$ represent the sentence sequence and the aspect sequence respectively. $N$ indicates the number of layers of the BERT encoder.


3 Dynamic Re-weighting BERT

In this section, we introduce the technical detail of DR-BERT. Specifically, we start with the problem definition, followed by an overall architecture of DR-BERT as illustrated in Figure 1.

**Problem Definition** In ABSA, a sentence-aspect pair $(S, A)$ is given. In this paper, the sentence is represented as $S = \{w_1^s, w_2^s, ..., w_{l_s}^s\}$ which consists of a series of $l_s$ words. The specific aspect is denoted as $A = \{w_1^a, w_2^a, ..., w_{l_a}^a\}$ which is a part of $S$. $l_a$ is the length of aspect words. The goal of ABSA is to learn a sentiment classifier that can precisely predict the sentiment polarity of sentence $S$ for specific aspect $A$. As the aspect-related information plays a key role in the prediction (Li et al., 2018; Zheng et al., 2020), this paper aims to dynamically select and encode the aspect-aware semantic information through the proposed model.

**Overall Architecture** DR-BERT mainly contains two components (i.e., BERT encoder and Dynamic Re-weighting Adapter), together with two modules (i.e., the embedding module and sentiment prediction module). The technical details of each part will be elaborated on as follows.

**3.1 Embedding Module**

To represent semantic information of the aspect words and context words better, we first map each word into a low-dimensional vector. Specifically, the inputs of DR-BERT are the sentence sequence and the corresponding aspect sequence. For the sentence sequence, we construct the BERT input as “[CLS]” + sentence + “[SEP]” and the sentence $S = \{w_1^s, w_2^s, ..., w_{l_s}^s\}$ can be transformed into the hidden states $s = \{s_i | i = 1, 2, ..., l_s\}$ with BERT embedding. For aspect sequences, we adopt the same method to get the representation vector of each word. Thus, through the embedding module, the aspect sequence $A = \{w_1^a, w_2^a, ..., w_{l_a}^a\}$ is mapped to $a^a = \{a_j | j = 1, 2, ..., l_a\}$. Note that, if the aspect sequence is a single word like “food”, the aspect representation is the embedding of the
single word “food”. While for the cases where the aspect sequence contains multiple words such as “system memory”, the aspect representation is the average of each word embedding (Sun et al., 2015). We can denote the aspect embedding process as:

\[
a = \begin{cases} 
    a_1, & \text{if } l_a = 1, \\
    (\sum_{j=1}^{l_a} a_j) / l_a, & \text{if } l_a > 1,
\end{cases}
\]

where \(a_j\) is the embedding of word \(j\) in the aspect sequence. \(a\) denotes the embedding of the aspect.

3.2 BERT Encoder

The architecture of BERT (Devlin et al., 2019) is akin to the Transformer (Vaswani et al., 2017). For simplicity, we omit some architecture details such as position encoding, layer normalization (Xu et al., 2019b) and residual connections (He et al., 2016).

1) Multi-head Self-attention Mechanism. In recent years, the multi-head self-attention mechanism (MultiHead) has received a wide range of applications in natural language processing. In the paper, we adopt MultiHead with \(h\) heads to obtain the overall semantics of the whole sentence. The product from each self-attention network is then concatenated and finally transformed as:

\[
m = \{m_i | i = 1, 2, \ldots, l_s\} = \text{MultiHead}(sW_h^Q, sW_h^K, sW_h^V),
\]

where \(h\) denotes the \(h\)-th attention head, \(W_h^Q, W_h^K\) and \(W_h^V\) are learnable parameters. Finally, the output feature is \(m = \{m_i | i = 1, 2, \ldots, l_s\}\). For detailed implementation of MultiHead, please refer to Transformer (Vaswani et al., 2017).

2) Position-wise Feed-Forward Network. Since the multi-head attention is a series of linear transformations, we then apply the position-wise feed-forward network (FFN) to learn the feature’s non-linear transformation. Specifically, the FFN consists of two linear transformations along with a ReLU activation in between. More formally:

\[
f = \{f_i | i = 1, 2, \ldots, l_s\} = \max(0, mW_1 + b_1)W_2 + b_2,
\]

where \(W_1, b_1, W_2\) and \(b_2\) are learnable parameters in the linear transformations.

So far, with the input \(S = \{w_1^s, w_2^s, \ldots, w_{l_s}^s\}\), we obtain the hidden states \(f = \{f_i | i = 1, 2, \ldots, l_s\}\) via the BERT encoder. Then, for the words’ hidden states of the sentence from FFN, we utilize the max-pooling operation to fairly select crucial features in the sentence (Lai et al., 2015; Zhang et al., 2019b), so as to obtain the original sentence representation \(h_s\) at the beginning of each re-weighting step:

\[
h_s = \text{Max_Pooling}(f_i | i = 1, 2, \ldots, l_s).
\]

3.3 Dynamic Re-weighting Adapter (DRA)

The currently attention mechanism in deep learning is essentially similar to the selective visual attention of human beings (Vaswani et al., 2017; You et al., 2016). However, as for the text semantic understanding, human brain will discover the intentional relationship of words at a sentential level (Taatgen et al., 2007; Sha et al., 2016; Sen et al., 2020) and link the incoming semantic information with pre-existing information stored within memory. Thus, we design a dynamic re-weighting adapter (DRA) which can dynamically emphasize the important aspect-aware words for the ABSA task.

As shown in the right part of Figure 1, based on overall semantics of the whole sentence, DRA further selects the most important word at each step with consideration of the specific aspect representation. Specifically, the inputs of DRA are the final outputs of the BERT encoder (i.e., \(h_s\)) and the original aspect embedding (i.e., \(a\)). In each step, we first utilize re-weighting attention to choose the word for current input from the input sequence \(\{s_i | i = 1, 2, \ldots, l_s\}\). Then, we utilize Gated Recurrent Unit (GRU)(Cho et al., 2014) to encode the chosen word and update the semantic representation of the review sentence.

Formally, we regard the calculation process as:

\[
a_t = F (s_1, s_2, \ldots, s_{l_s}, h_{t-1}, a), \\
h_t = \text{GRU} (a_t, h_{t-1}), \quad t \in [1, T]
\]

where \(a\) is the original embedding vector of the aspect words. \(a_t\) is the output of re-weighting function \(F\). \(T\) denotes the dynamic re-weighting length over the sentences, which represents the cognitive threshold of human beings. \(h_0 = h_s\) is the initial state and \(h_T\) is the output hidden states of DRA.

1) The Re-weighting Function. More specifically, we utilize the attention mechanism to achieve the re-weighting function \(F\), which aims to select the most important aspect-related word at each step. The calculation can be formulated as:

\[
S = [s_1, s_2, \ldots, s_{l_s}], \\
M = W_s S + (W_a h_{t-1} + W_a a) \otimes w, \\
m = \omega^T \tanh(M),
\]

where \(W_s, W_a, \omega\) are learnable parameters.
where $S$ denotes the original sentence embedding, $M$ is the fusion representation of the aspects and the sentences. $W_s$, $W_d$, $W_a$ and $\omega$ are trainable parameters. $w \in \mathbb{R}^{l_s}$ is a row vector of 1 and $\otimes$ denotes the outer product.

Subsequently, to better encode aspect-aware semantics, we choose the most important word (i.e., one word) at each step for the specific aspect.

$$
\alpha_i = \frac{\exp(m_i)}{\sum_{k=1}^{l_s} \exp(m_k)}, \quad a_i = s_j, (j = \text{Index}(\max(\alpha_i)))
$$

where $m_i$ and $\alpha_i$ are the hidden state and the attention score of $i$-th word in the sentence. $a_i$ is the chosen word which is most related to the specific aspect at $t$-th step. However, Index($\max()$) operation has no derivative, which means its gradient could not be calculated. Inspired by softmax function, we modify the Eq.7 and employ the following operation to re-weight the contextual words:

$$
a_t = \sum_{i=1}^{l_s} \frac{\exp(\lambda m_i)}{\sum_{k=1}^{l_s} \exp(\lambda m_k)} s_i.
$$

Note that, we design a hyper-parameter $\lambda$ to ensure our model achieves the above purpose. Specifically, the softmax function can exponentially increase or decrease the signal, thereby highlighting the information we want to enhance. Thus, when $\lambda$ is an arbitrarily large value, the attention score of the chosen word is infinitely close to 1, and other words are infinitely close to 0. In this way, the most important word (i.e., one word) will be extract from the context at each re-weighting step.

2) The GRU Function. To better encode semantic of the whole sentence, we also employ GRU to further imitate the procedure of human semantic comprehension under the specific context, which is consistent with the process of people adjusting to a new text based on their understanding behavior. Therefore, given a previous vector embedding, the hidden vectors of GRU are calculated by receiving it as input:

$$
z_t = \sigma(W_z \cdot [h_{t-1}, a_t])
$$

$$
r_t = \sigma(W_r \cdot [h_{t-1}, a_t])
$$

$$
h_t = \tanh(W \cdot [r_t \otimes h_{t-1}, a_t])
$$

where $\sigma$ is the logistic sigmoid function. $z_t$ and $r_t$ denote the update gate and reset gate respectively at the time step $t$.

### 3.4 Sentiment Predicting

After applying BERT layers and DRA on the input sentence, its root representation (i.e., $s$) is convert into the feature representation $e$:

$$
e = \{e_i | i = 1, 2, \ldots, l_s\} = (W_e f + U_e h_T + b_e),
$$

where $W_e$, $U_e$ and $b_e$ are trainable parameters. After $N$-th stacked BERT layers, we obtain the final representation of the sentence (i.e., $e_N$). Then, we feed it into a Multilayer Perceptron (MLP) and map it to the probabilities over the different sentiment polarities via a softmax layer:

$$
R_l = \text{Relu}(W_l R_{l-1} + b_l), \quad y = \text{softmax}(W_o R_h + b_o), \quad l \in [1, h_l],
$$

where $W_l$, $W_o$, $b_l$ and $b_o$ are learned parameters. $R_l$ is the hidden state of $l$-th layer MLP ($R_0 = e_N$, $l \in [1, h_l]$). $R_h$ is the state of final layer which is also regard as the output of the MLP. $y$ is the predicted sentiment polarity distribution.

### 3.5 Model Training

Finally, we applies the cross-entropy loss function for model training:

$$
\mathcal{L} = - \sum_{i=1}^{M} \sum_{j=1}^{C} y^j_i \log(y^j_i) + \beta \|\Theta\|_2^2,
$$

where $y^j_i$ is the ground truth sentiment polarity. $C$ is the number of labels (i.e, 3 in our task). $M$ is the number of training samples. $\Theta$ corresponds to all of the trainable parameters.

### 4 Experiment

#### 4.1 Datasets

We mainly conduct experiments on three benchmark ABSA datasets, including “Laptop”, “Restaurant” (Pontiki et al., 2014) and “Twitter” (Dong et al., 2014). Each data item is labeled with three

<table>
<thead>
<tr>
<th>Datasets</th>
<th>#Positive</th>
<th>#Negative</th>
<th>#Neural</th>
<th>#L</th>
<th>#M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>Test</td>
<td>Train</td>
<td>Test</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Restaurant</td>
<td>2164 728</td>
<td>807 196</td>
<td>637 196</td>
<td>20</td>
<td>45.5</td>
</tr>
<tr>
<td>Laptop</td>
<td>994 341</td>
<td>870 128</td>
<td>464 169</td>
<td>19</td>
<td>36.5</td>
</tr>
<tr>
<td>Twitter</td>
<td>1561 173</td>
<td>1560 173</td>
<td>3127 346</td>
<td>16</td>
<td>10.2</td>
</tr>
</tbody>
</table>

Table 1: The statistics of three benchmark datasets. #L is the average length of sentences. #M is the proportion (%) of samples with multiple (i.e., more than 1) aspects.
sentiment polarities (i.e., positive, negative and neutral). The statistics of the datasets are presented in Table 1. Moreover, we follow the dataset configurations of previous studies strictly. For all datasets, we randomly sample 10% items from the training set and regard them as the development set.

4.2 Hyperparameters Settings

In the implementation, we build our framework based on the official bert-base models ($n_{layers}$=12, $n_{heads}$=12, $n_{hidden}$=768). The hidden size of GRUs and re-weighting length of DRA are set to 256 and 7. The learning rate is tuned amongst $[2e-5, 5e-5$ and $1e-3]$ and the batch size is manually tested in $[16, 32, 64, 128]$. The dropout rate is set to 0.2. The hyper-parameter $l$, $\beta$ and $\lambda$ have been carefully adjusted, and final values are set to 3, 0.8 and 100 respectively. The model is trained using the Adam optimizer and evaluated by two widely used metrics. The parameters of baseline models are in accordance with the default configuration of the original paper. We run our model three times with different seeds and report the average performance.

4.3 Baselines

- **Attention-based Models**: MemNet (Tang et al., 2016), ATAE-LSTM (Wang et al., 2016), IAN (Ma et al., 2017), AOA (Huang et al., 2018), MGNNet (Fan et al., 2018), TNet (Li et al., 2018).
- **Pre-trained Models**: Fine-tune BERT (Devlin et al., 2019), BERT-PT (Xu et al., 2019a), BERT-SPC, AEN-BERT (Song et al., 2019), RGAT-BERT (Wang et al., 2020), T-GCN (Tian et al., 2021).

The baseline methods have comprehensive coverage of the recent related SOTA models recently. Most of them are detailed in section 2.1. For space-saving, we do not detail them in this section.

4.4 Experimental Results

From the results in Table 2, we have the following observations. First, BERT-based methods beat most of the attention-based methods (e.g., IAN and TNet) in both metrics. The phenomenon indicates the powerful ability of the pre-trained language models. That is also why we adopt BERT as base encoder to learn the overall semantic representation of the whole sentences.

Second, by comparing non-specific BERT models (i.e., BERT and BERT-PT) with task-specific models (e.g., RGAT-BERT) for ABSA, we find that the task-specific BERT models perform better than the non-specific models. Specifically, we can also observe the performance trend that T-GCN>RGAT-BERT>AEN-BERT>BERT-PT>BERT, which is consistent with the previous assumption that aspect-related information is the crucial influence factor for the performance of the ABSA model.

Finally, despite the outstanding performance of previous models, our DR-BERT still outperforms the most advanced baseline (i.e., T-GCN or RGAT-BERT) no matter in terms of Accuracy or F1-score. The results demonstrate the effectiveness of the dynamic modeling strategy based on the procedure of semantic comprehension. Meantime, it also indicates that our proposed DRA can better grasp the aspect-aware semantics of the sentence than other BERT plus-in components in previous methods.
Table 3: The ablation study on different components which conducted on the test set of the Laptop dataset. “BERT-Base” indicates the vanilla BERT. “+” indicates the setting with plus-in components.

<table>
<thead>
<tr>
<th>Model Variants</th>
<th>Laptop</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accuracy</td>
</tr>
<tr>
<td>BERT-Base</td>
<td>77.29</td>
</tr>
<tr>
<td>(1): + MLP</td>
<td>77.94</td>
</tr>
<tr>
<td>(2): + DRA</td>
<td>80.66</td>
</tr>
<tr>
<td>(3): + DRA on top 3 layers</td>
<td>78.64</td>
</tr>
<tr>
<td>(4): + DRA on top 6 layers</td>
<td>79.17</td>
</tr>
<tr>
<td>(5): + DRA on top 9 layers</td>
<td>80.22</td>
</tr>
<tr>
<td>(6): DR-BERT</td>
<td>81.45</td>
</tr>
</tbody>
</table>

Table 3, we study the influence of different components in our framework, including the DRA and MLPs. We can find that without utilizing adapters and MLPs, DR-BERT degenerates into the BERT model, which gains the worst performance among all the variants. The phenomenon indicates the effective of the DRA and MLP modules. Moreover, through comparing (1) and (2), we can easily conclude that DRA plays a more crucial role in the final sentiment prediction than MLPs.

Since BERT models are usually quite deep (e.g., 12 layers), we only insert the dynamic re-weighting adapter into top layers (i.e., 3-th, 6-th, and 9-th layers) to further verify the effectiveness of the DRA module. The results are shown in Table 3 (3), (4), and (5). We observe that when introducing adapters to the top layers of DR-BERT, our framework still outperforms the BERT model, showing that the DRA is efficient in encoding the aspect-aware semantics over the whole sentence. In addition, we can also find that the more adapter incorporated in BERT layers the higher performance gained, illustrating the importance of modeling the deep dynamic semantics over the sentence.

4.5 Ablation Study

Ablations on the Proposed Components. In this subsection, we investigate the influence of the scale of adapters on different datasets. As shown in Figure 2, we tune the adapter’s dynamic re-weighting length ($T$) in a wide range (i.e., 2 to 10). Specifically, the performance of DR-BERT first becomes better with the increasing of re-weighting length and achieving the best result at around 7. Then, as the length continues to increase, the performance continues to decline. This phenomenon is consistent with the psychological findings that human memory focuses on nearly seven words (Tononi, 2008; Koch and Tsuchiya, 2007), which further indicates the effectiveness of DRA in modeling human-like (dynamic) semantic comprehension.

Besides, compared with the best-performed baseline (blue lines), our model can achieve better performance with only 4 or 5 times of re-weighting at most test sets, illustrating the efficiency of the re-weighting adapter. On the other hand, we can also find that DR-BERT always gives superior performance compared to the BERT-based model (green lines), even with the lowest re-weighting length. All those results show that DR-BERT could better comprehend aspect-aware dynamic semantics in aspect-based sentiment analysis.

4.6 Interpretability Verification

Comparison of Semantic Comprehension. To evaluate model rationality and interpretability, we conduct an study for dynamic semantic comprehension by eye tracker. As shown in Figure 3 (a),

While the $20 entree range is not overly expensive, in New York City, there is definitely better food in that range, and so Sophie, despite it is lovely atmosphere, will most likely not be a restaurant to which I will return.

(a) Human Cognition

(b) DRA Chosen Words

Figure 3: Comparison of the semantic understanding process between human reading and DRA when judging the sentiment polarity of aspect “food”. (a) is the visualization of the human understanding process from the eye tracker. (b) denotes aspect-aware words from re-weighting function.

The procedure of human semantic comprehension is generated by the eye tracker: https://www.tobiipro.com/product-listing/nano/
when a person tries to understand a relatively long sentence, he/she first read the entire sentence. Subsequently, after giving a specific aspect, he/she will dynamically select related words based on the previous memory state until he/she fully understands the sentiment polarity of the given aspect.

Interestingly, the above phenomenon is consistent with our dynamic re-weighting adapter’s chosen result. Specifically, as Figure 3 (b) shows, with the re-weighting function $F$ (i.e., Equation 5 and 6), our model dynamically choose the words “food, better, while, definitely, not, …”, which have proven to be very important for predicting the sentiment of aspect “food” in Figure 3 (a). Those experimental results again fully indicate the effectiveness and interpretability of our proposed model in dynamic learning aspect-aware information.

**The Influence of multiple Aspects.** As aspect-related information plays a key role in ABSA and at least 10.2% of reviews contain multiple aspects as shown in Table 1, we are curious about the model’s performance in the complex scenarios, e.g., a review sentence contains multiple aspects. Therefore, we randomly choose an example to explore how the selection of the context words will correspondingly change with different inputs. The visualization results are shown in Figure 4. Specifically, the chosen sentence has three different aspects with their sentiment polarity, i.e., “System memory”-negative, “DDR5”-positive and “DDR3”-negative. Take the aspect “DDR5” as example, it is positive which is contrary to “DDR3”. After receiving the overall semantic of the whole sentence, readers tend to associate “DDR5” with the context words (“would”, “have”) to predict the correct sentiment “positive”. For other two aspects, the observations are consistent with “DDR3”. In summary, all those results show that DR-BERT could dynamically extract the vital information to achieve aspect-aware semantic understanding even in a more complex scenario.

4.7 Error Analysis

Table 4 displays three review examples and their prediction results by BERT, RGAT-BERT, and our DR-BERT. As we can see from the “BERT-base” column, when there are multiple aspects, the vanilla BERT often makes the wrong classification since it tends to learn the overall sentiment polarity of the sentences instead of the aspect-aware semantic understanding even in a more complex scenario.

### Table 4: Error analysis of two review item from laptop and restaurant. The colored words in brackets represents ground truth sentiment label of the corresponding aspect. The symbol ✓ means the predicting sentiment is correct, and the other symbol means the predicting sentiment is wrong.

<table>
<thead>
<tr>
<th>Case Examples</th>
<th>BERT-base</th>
<th>RGAT-BERT</th>
<th>DR-BERT</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Aspects:</strong> “system memory” (Neg.), “DDR5” (Pos.), “DDR3” (Neg.)</td>
<td>Pos/Neg/Neg</td>
<td>Neg/Pos/Pos</td>
<td>Neg/Pos/Neg</td>
</tr>
<tr>
<td><strong>Sentence:</strong> It could be a perfect laptop if it would have faster system memory and its radeon would have DDR5 instead of DDR3.</td>
<td>✓/✓/✗</td>
<td>✓/✓/✗</td>
<td>✓/✓/✓</td>
</tr>
<tr>
<td><strong>Aspects:</strong> “Supplied software” (Neu.), “software” (Pos.), “Windows” (Neg.)</td>
<td>Pos/Pos/Pos</td>
<td>Pos/Pos/Neu</td>
<td>Pos/Pos/Neg</td>
</tr>
<tr>
<td><strong>Sentence:</strong> Supplied software: The software that comes with this machine is greatly welcomed compared to what Windows comes with.</td>
<td>✓/✓/✓</td>
<td>✓/✓/✗</td>
<td>✓/✓/✓</td>
</tr>
<tr>
<td><strong>Aspects:</strong> “waiter” (Neg.), “served” (Neg.), “specials” (Pos.)</td>
<td>Neg/Neg/Neg</td>
<td>Neg/Neg/Neu</td>
<td>Neg/Neg/Pos</td>
</tr>
<tr>
<td><strong>Sentence:</strong> First, the waiter who served us neglected to fill us in on the specials, which I would have chosen had I known about them.</td>
<td>✓/✓/✓</td>
<td>✓/✓/✗</td>
<td>✓/✓/✓</td>
</tr>
</tbody>
</table>

The attention weight analysis is detailed in Appendix A.1.
helpful syntactic dependency relations. Therefore, RGAT-BERT makes a wrong sentiment prediction. However, our DR-BERT model, succeeding in predicting most sentiment labels by considering the dynamic changing of the aspect-aware semantic. For other two case examples, the observations are consistent. Note that, for aspect “Supplied software” in second sentence, two overlap aspects appear in the same sentence makes it more difficult to distinguish the different sentiment between them. Thus, precisely determine its sentiment polarity is a big challenge for human, let alone deep learning models. This also leaves space for future exploration.

5 Computation Time Comparison

We also compared the computation runtime of three baseline methods. All of the models are performed on a Linux server with 64 Intel(R) CPUs and 4 Tesla V100 32GB GPUs. From the results shown in Table 5, we can first observe that the training time of a single epoch in DR-BERT performs better than T-GCN, which is based on GCN. Meanwhile, the training time of all these BERT-based models is similar (i.e., there is no significant difference). The possible reason is that the official datasets are small, and it is hard to influence the overall runtime of PLMs with such a small amount of data. Second, compared with other models, the training time of the ATAE-LSTM model is less (always an order of magnitude lower). For example, the ATAE-LSTM only needs about two minutes to achieve optimal performance in the restaurant dataset, while BERT-based models require more than 26 minutes. Therefore, though DR-BERT contains a Dynamic Re-weighting adapter based on GRU, the computation time is much lower than the BERT-based framework. In summary, the observations above show that the computation time of our DR-BERT model is within an acceptable range.

6 Conclusion and Future Works

This paper introduced a new approach named Dynamic Re-weighting BERT (DR-BERT) for aspect-based sentiment analysis. Specifically, we first employed the BERT layers as a base encoder to learn the overall semantic features of the whole sentence. Then, inspired by human semantic comprehension, we devised a new Dynamic Re-weighting Adapter (DRA) to enhance aspect-aware semantic features in the sentiment learning process. In addition, we inserted the DRA into the BERT layers to address the limitations of the vanilla pre-trained model in ABSA task. Extensive experiments on three benchmark datasets demonstrated the effectiveness and interpretability of the proposed model, with good semantic comprehension insights for future nature language modeling. Moreover, the error analysis was performed on incorrectly predicted examples, leading to some insights into the ABSA task.

We hope our research can help boost excellent work for aspect-based sentiment analysis from different perspectives. In the future, we plan to extend our method to other tasks like Sentence Semantic Matching, Relation Extraction, etc., which can also benefit from utilizing the dynamic semantics. Besides, we will explore whether DR-BERT can make any positive changes based on previous mistakes during the dynamic semantic understanding.

7 Acknowledgments

We would like to thank the anonymous reviewers for the helpful comments. This research was partially supported by grants from the National Key R&D Program of China (No. 2021YFF0901003), and the National Natural Science Foundation of China (No. 61922073, 61727809, 62006066 and 72101176). We appreciate all the authors for their fruitful discussions. We also special thanks to all the first-line healthcare providers that are fighting the war of COVID-19.
References


Kai Zhang, Qi Liu, Hao Qian, Biao Xiang, Qing Cui, Jun Zhou, and Enhong Chen. 2021a. Eatn: An efficient adaptive transfer network for aspect-level sentiment analysis. IEEE Transactions on Knowledge and Data Engineering.


Appendix

A Attention Visualization

In order to better quantify and visualize why the current attention or pre-trained model fails, we visualize the attention weight in different models. In the following, we report the visualization results of these models and relative analysis w.r.t the aspect-based sentiment predicting performance.

A.1 Self-Attentions of Aspect in BERT

Intuitively, self-attention can serve as a way to aggregate the representations of contextual tokens into an aspect. Following (Clark et al., 2019; Xu et al., 2020), we notice some attention heads exhibit general patterns. As shown in Figure 5 (a), the sentiment polarity of aspect “system memory” is mistakenly predicted into positive by the BERT-base model. The reason may lie in that most of the attention heads focus on the explicit sentiment word (“faster”) while largely ignoring words (“would”, “have”) that may cause sentiment reversal.

Besides, for aspects “DDR5” and “DDR3”, the BERT-base model assigns more attention weight to the token “[CLS]”, representing the whole sentence’s overall semantic. Therefore, both aspects’ sentiment polarity is classified into negative, which is the sentiment polarity of the entire sentence. As mentioned previously (see Section 1), the visualization result from Figure 5 (b) and (c) further indicates the weakness of the vanilla BERT; that is, it can not fully understand the local semantics of sentences by considering the given aspect.

A.2 Attentions of Aspect in DR-BERT

In addition to the visualization results and analysis of the BERT-base model, we also visualize the attention weight in DR-BERT layers to intuitively demonstrate it’s effects and interpretability.

Specifically, compare with Figure 6 (a) and Figure 5 (a), we can observe that DR-BERT pays more attention to the words “would” and “have”. This is possible because that DRA can select crucial aspect-aware words dynamically, thus enforcing the BERT encoder to allocate higher weight to those words. Obviously, by exploring the semantic of those two words, the sentiment polarity of aspect “system memory” can be precisely predicted.

Besides, in contrast with the BERT-base model, the attention weight produced by DR-BERT concentrates in a small region for different aspects, as shown in Figure 6 (b) and (c). More specifically, words more related to “DDR3” are positioned in “instead” and “of” rather than “[CLS]” (denotes as the global semantic in BERT), which makes the semantic of the prediction process more concentrated on a specific aspect (“DDR3”). We can find the same phenomenon on the aspect “DDR5” in Figure 6 (b). In conclusion, the main factor that determines the modeling of aspect-aware semantics is DRA, which effectively characterizes the dynamic semantic changes in ABSA.

B Generality Discussion

To the best of our knowledge, our framework is the first work to jointly integrate human cognition process with pre-trained models from the perspective of both deep learning and cognitive psychology. Unlike most previous studies that directly utilize pre-trained BERT as a feature extractor, we incorporate it with a novel dynamic re-weighting adapter that can further exploit aspect-aware dynamic semantics by imitating human cognitive processes.

Comparing with the related works that also utilize pre-trained models, our method is more general because it is straightforward to transform into traditional fine-tuning BERT by setting the re-weighting length to 0. Comparing with the relevant works in section 2, our method is more comprehensive because we fully explore the potential of the incorporation between the human cognitive process and the pre-trained model. In addition, we can easily extend the architectures to adjust to different downstream NLP tasks. For example, in the semantic matching task, the framework can transform into a suitable encoder by considering one sentence as BERT input and another sentence as DRA input.

C Further Experiment Analysis

In order to answer the concern of the anonymous reviewer “Do you try the BERT-Large setting? I’m wondering whether there is the same performance trend as in the BERT-Base setting and adding the results of the BERT-Large version will be more convincing to show the effectiveness of the DRA.”, we conducted a more extensive experiment based on the BERT-large model6 to verify the effect of the proposed model and the DRA module. The specific performance is shown in Table 6.

---

6The pre-trained BERT-large-uncased model was downloaded from the HuggingFace: https://huggingface.co/bert-large-uncased
Figure 5: The attention-head view visualizes attention in one or more heads in the BERT-base model. The lines show the attention from each token (left) to every other token (right). Darker lines indicate higher attention weights. The colors correspond to different attention heads.

Figure 6: The visualization of attention-heads in the proposed DR-BERT. As we can see from (b) and (c), compare with the visualization in Figure 5 (b) and (c), each aspect of attention is more concentrated in a small region.

<table>
<thead>
<tr>
<th>Category</th>
<th>Methods</th>
<th>Laptop</th>
<th>Restaurant</th>
<th>Twitter</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accuracy</td>
<td>F1-score</td>
<td>Accuracy</td>
<td>F1-score</td>
</tr>
<tr>
<td></td>
<td>RGAT-BERT (Wang et al., 2020)</td>
<td>78.21</td>
<td>74.07</td>
<td>86.60</td>
</tr>
<tr>
<td></td>
<td>T-GCN (Tian et al., 2021)</td>
<td>80.88</td>
<td>77.03</td>
<td>86.16</td>
</tr>
<tr>
<td>Ours.</td>
<td>DR-BERT</td>
<td>81.45</td>
<td>78.16</td>
<td>87.72</td>
</tr>
<tr>
<td>Ours.</td>
<td>DR-BERT-large</td>
<td>82.18</td>
<td>78.93</td>
<td>88.17</td>
</tr>
</tbody>
</table>

Table 6: Experimental results (%) in three benchmark datasets. We underline the best performed baseline method. To further demonstrate the effectiveness of DRA, we construct a more advanced DR-BERT (i.e., DR-BERT-large) by using the BERT-large-uncased model.