Attentional Encoder Network for Targeted Sentiment Classification

Youwei Song, Jiahai Wang *, **Tao Jiang, Zhiyue Liu, Yanghui Rao**
School of Data and Computer Science
Sun Yat-sen University
Guangzhou, China
{songyw5,jiangt59,liuzhy93}@mail2.sysu.edu.cn
{wangjiah,raoyangh}@mail.sysu.edu.cn

Abstract

Targeted sentiment classification aims at determining the sentimental tendency towards specific targets. Most of the previous approaches model context and target words with RNN and attention. However, RNNs are difficult to parallelize and truncated backpropagation through time brings difficulty in remembering long-term patterns. To address this issue, this paper proposes an Attentional Encoder Network (AEN) which eschews recurrence and employs attention based encoders for the modeling between context and target. We raise the label unreliability issue and introduce label smoothing regularization. We also apply pre-trained BERT to this task and obtain new state-of-the-art results. Experiments and analysis demonstrate the effectiveness and lightweight of our model. [1]

1 Introduction

Targeted sentiment classification is a fine-grained sentiment analysis task, which aims at determining the sentiment polarities (e.g., negative, neutral, or positive) of a sentence over "opinion targets" that explicitly appear in the sentence. For example, given a sentence *"I hated their service, but their food was great"*, the sentiment polarities for the target *"service"* and *"food"* are negative and positive respectively. A target is usually an entity or an entity aspect.

In recent years, neural network models are designed to automatically learn useful low-dimensional representations from targets and contexts and obtain promising results (Dong et al., 2014; Tang et al., 2016a). However, these neural network models are still in infancy to deal with the fine-grained targeted sentiment classification task.

Attention mechanism, which has been successfully used in machine translation (Bahdanau et al., 2014), is incorporated to enforce the model to pay more attention to context words with closer semantic relations with the target. There are already some studies use attention to generate target-specific sentence representations (Wang et al., 2016; Ma et al., 2017; Chen et al., 2017) or to transform sentence representations according to target words (Li et al., 2018). However, these studies depend on complex recurrent neural networks (RNNs) as sequence encoder to compute hidden semantics of texts.

The first problem with previous works is that the modeling of text relies on RNNs. RNNs, such as LSTM, are very expressive, but they are hard to parallelize and backpropagation through time (BPTT) requires large amounts of memory and computation. Moreover, essentially every training algorithm of RNN is the truncated BPTT, which affects the model's ability to capture dependencies over longer time scales (Werbos, 1990). Although LSTM can alleviate the vanishing gradient problem to a certain extent and thus maintain long distance information, this usually requires a large amount of training data. Another problem that previous studies ignore is the label unreliability issue, since *neutral* sentiment is a fuzzy sentimental state and brings difficulty for model learning. As far as we know, we are the first to raise the label unreliability issue in the targeted sentiment classification task.

中立情感很模糊.

This paper propose an attention based model to solve the problems above. Specifically, our model eschews recurrence and employs attention as a competitive alternative to draw the introspective and interactive semantics between target and context words. To deal with the label unreliability issue, we employ a label smoothing regularization to encourage the model to be less confident with

反省的,内省的.

---
*The corresponding author.

[1]Source code is available at https://github.com/songyouwei/ABSA-PyTorch/tree/aen.

fuzzy labels. We also apply pre-trained BERT (Devlin et al., 2018) to this task and show our model enhances the performance of basic BERT model. Experimental results on three benchmark datasets show that the proposed model achieves competitive performance and is a lightweight alternative of the best RNN based models.

The main contributions of this work are presented as follows:

1. We design an attentional encoder network to draw the hidden states and semantic interactions between target and context words.
2. We raise the label unreliability issue and add an effective label smoothing regularization term to the loss function for encouraging the model to be less confident with the training labels.
3. We apply pre-trained BERT to this task, our model enhances the performance of basic BERT model and obtains new state-of-the-art results.
4. We evaluate the model sizes of the compared models and show the lightweight of the proposed model.

2 Related Work

The research approach of the targeted sentiment classification task including traditional machine learning methods and neural networks methods.

Traditional machine learning methods, including rule-based methods (Ding et al., 2008) and statistic-based methods (Jiang et al., 2011), mainly focus on extracting a set of features like sentiment lexicons features and bag-of-words features to train a sentiment classifier (Rao and Ravichandran, 2009). The performance of these methods highly depends on the effectiveness of the feature engineering works, which are labor intensive.

In recent years, neural network methods are getting more and more attention as they do not need handcrafted features and can encode sentences with low-dimensional word vectors where rich semantic information stained. In order to incorporate target words into a model, Tang et al. (2016a) propose TD-LSTM to extend LSTM by using two single-directional LSTM to model the left context and right context of the target word respectively. Tang et al. (2016b) design MemNet which consists of a multi-hop attention mechanism with an external memory to capture the importance of each context word concerning the given target. Multiple attention is paid to the memory represented by word embeddings to build higher semantic information. Wang et al. (2016) propose ATAE-LSTM which concatenates target embeddings with word representations and let targets participate in computing attention weights. Chen et al. (2017) propose RAM which adopts multiple-attention mechanism on the memory built with bidirectional LSTM and nonlinearly combines the attention results with gated recurrent units (GRUs). Ma et al. (2017) propose IAN which learns the representations of the target and context with two attention networks interactively.

3 Proposed Methodology

Given a context sequence $\mathbf{w^c} = \{w_1^c, w_2^c, ..., w_n^c\}$ and a target sequence $\mathbf{w^t} = \{w_1^t, w_2^t, ..., w_m^t\}$, where $\mathbf{w^t}$ is a sub-sequence of $\mathbf{w^c}$. The goal of this model is to predict the sentiment polarity of the sentence $\mathbf{w^c}$ over the target $\mathbf{w^t}$.

Figure 1 illustrates the overall architecture of the proposed **A**ttentional **E**ncoder **N**etwork (AEN), which mainly consists of an embedding layer, an attentional encoder layer, a target-specific attention layer, and an output layer. Embedding layer has two types: GloVe embedding and BERT embedding. Accordingly, the models are named **AEN-GloVe** and **AEN-BERT**.

3.1 Embedding Layer

3.1.1 GloVe Embedding

Let $L \in \mathbb{R}^{d_{emb} \times |V|}$ to be the pre-trained GloVe (Pennington et al., 2014) embedding matrix, where d_{emb} is the dimension of word vectors and $|V|$ is the vocabulary size. Then we map each word $w^i \in \mathbb{R}^{|V|}$ to its corresponding embedding vector $e_i \in \mathbb{R}^{d_{emb} \times 1}$, which is a column in the embedding matrix L.

3.1.2 BERT Embedding

BERT embedding uses the pre-trained BERT to generate word vectors of sequence. In order to facilitate the training and fine-tuning of BERT model, we transform the given context and target to "[CLS] + context + [SEP]" and "[CLS] + target + [SEP]" respectively.

3.2 Attentional Encoder Layer

The attentional encoder layer is a parallelizable and interactive alternative of LSTM and is applied

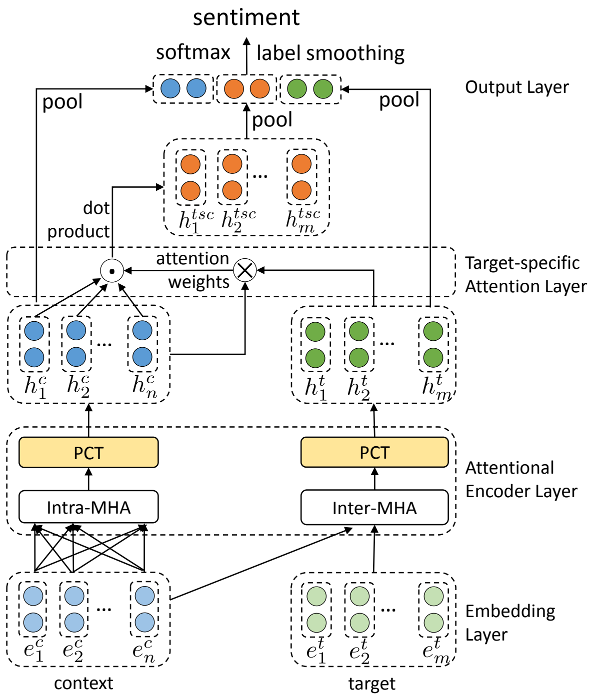

Figure 1: Overall architecture of the proposed AEN.

to compute the hidden states of the input embeddings. This layer consists of two submodules: the Multi-Head Attention (MHA) and the Point-wise Convolution Transformation (PCT).

3.2.1 Multi-Head Attention

Multi-Head Attention (MHA) is the attention that can perform multiple attention function in parallel. Different from Transformer (Vaswani et al., 2017), we use **Intra-MHA** for introspective context words modeling and **Inter-MHA** for context-perceptive target words modeling, which is more lightweight and target is modeled according to a given context.

An attention function maps a key sequence $\mathbf{k} = \{k_1, k_2, ..., k_n\}$ and a query sequence $\mathbf{q} = \{q_1, q_2, ..., q_m\}$ to an output sequence $\mathbf{o}$:

$$Attention(\mathbf{k}, \mathbf{q}) = softmax(f_s(\mathbf{k}, \mathbf{q}))\mathbf{k} \quad (1)$$

where f_s denotes the alignment function which learns the semantic relevance between q_j and k_i:

$$f_s(k_i, q_j) = tanh([k_i; q_j] \cdot W_{att}) \quad (2)$$

where $W_{att} \in \mathbb{R}^{2d_{hid}}$ are learnable weights.

MHA can learn *n_head* different scores in parallel child spaces and is very powerful for alignments. The n_{head} outputs are concatenated and projected to the specified hidden dimension d_{hid}, namely,

$$MHA(\mathbf{k}, \mathbf{q}) = [\mathbf{o}^1; \mathbf{o}^2 ...; \mathbf{o}^{n_{head}}] \cdot W_{mh} \quad (3)$$

$$\mathbf{o}^h = Attention^h(\mathbf{k}, \mathbf{q}) \quad (4)$$

where ";" denotes vector concatenation, $W_{mh} \in \mathbb{R}^{d_{hid} \times d_{hid}}$, $\mathbf{o}^h = \{o_1^h, o_2^h, ..., o_m^h\}$ is the output of the h-th head attention and $h \in [1, n_{head}]$.

Intra-MHA, or multi-head self-attention, is a special situation for typical attention mechanism that $\mathbf{q} = \mathbf{k}$. Given a context embedding $\mathbf{e^c}$, we can get the introspective context representation $\mathbf{c^{intra}}$ by:

$$\mathbf{c^{intra}} = MHA(\mathbf{e^c}, \mathbf{e^c}) \quad (5)$$

The learned context representation $\mathbf{c^{intra}} = \{c_1^{intra}, c_2^{intra}, ..., c_n^{intra}\}$ is aware of long-term dependencies.

Inter-MHA is the generally used form of attention mechanism that $\mathbf{q}$ is different from $\mathbf{k}$. Given a context embedding $\mathbf{e^c}$ and a target embedding $\mathbf{e^t}$, we can get the context-perceptive target representation $\mathbf{t^{inter}}$ by:

$$\mathbf{t^{inter}} = MHA(\mathbf{e^c}, \mathbf{e^t}) \quad (6)$$

After this interactive procedure, each given target word e_j^t will have a composed representation selected from context embeddings $\mathbf{e^c}$. Then we get the context-perceptive target words modeling $\mathbf{t^{inter}} = \{t_1^{inter}, t_2^{inter}, ..., t_m^{inter}\}$.

3.2.2 Point-wise Convolution Transformation

A **P**oint-wise **C**onvolution **T** ransformation (PCT) can transform contextual information gathered by the MHA. Point-wise means that the kernel sizes are 1 and the same transformation is applied to every single token belonging to the input. Formally, given a input sequence $\mathbf{h}$, PCT is defined as:

$$PCT(\mathbf{h}) = \sigma(\mathbf{h} * W_{pc}^1 + b_{pc}^1) * W_{pc}^2 + b_{pc}^2 \quad (7)$$

where σ stands for the ELU activation, $*$ is the convolution operator, $W_{pc}^1 \in \mathbb{R}^{d_{hid} \times d_{hid}}$ and $W_{pc}^2 \in \mathbb{R}^{d_{hid} \times d_{hid}}$ are the learnable weights of the two convolutional kernels, $b_{pc}^1 \in \mathbb{R}^{d_{hid}}$ and $b_{pc}^2 \in \mathbb{R}^{d_{hid}}$ are biases of the two convolutional kernels.

Given $\mathbf{c^{intra}}$ and $\mathbf{t^{inter}}$, PCTs are applied to get the output hidden states of the attentional encoder layer $\mathbf{h^c} = \{h_1^c, h_2^c, ..., h_n^c\}$ and $\mathbf{h^t} =$

$\{h_1^t, h_2^t, ..., h_m^t\}$ by:

$$\mathbf{h^c} = PCT(\mathbf{c^{intra}}) \tag{8}$$

$$\mathbf{h^t} = PCT(\mathbf{t^{inter}}) \tag{9}$$

3.3 Target-specific Attention Layer

After we obtain the introspective context representation $\mathbf{h^c}$ and the context-perceptive target representation $\mathbf{h^t}$, we employ another MHA to obtain the target-specific context representation $\mathbf{h^{tsc}} = \{h_1^{tsc}, h_2^{tsc}, ..., h_m^{tsc}\}$ by:

$$\mathbf{h^{tsc}} = MHA(\mathbf{h^c}, \mathbf{h^t}) \tag{10}$$

The multi-head attention function here also has its independent parameters.

3.4 Output Layer

We get the final representations of the previous outputs by average pooling, concatenate them as the final comprehensive representation $\tilde{\mathbf{o}}$, and use a full connected layer to project the concatenated vector into the space of the targeted C classes.

$$\tilde{\mathbf{o}} = [h_{avg}^c; h_{avg}^t; h_{avg}^{tsc}] \tag{11}$$

$$x = \tilde{W_o}^T \tilde{\mathbf{o}} + \tilde{b_o} \tag{12}$$

$$y = softmax(x) \tag{13}$$

$$= \frac{exp(x)}{\sum_{k=1}^{C} exp(x)} \tag{14}$$

where $y \in \mathbb{R}^C$ is the predicted sentiment polarity distribution, $\tilde{W_o} \in \mathbb{R}^{1 \times C}$ and $\tilde{b_o} \in \mathbb{R}^C$ are learnable parameters.

3.5 Regularization and Model Training

Since *neutral* sentiment is a very fuzzy sentimental state, training samples which labeled *neutral* are unreliable. We employ a **L**abel **S**moothing **R**egularization (LSR) term in the loss function. which penalizes low entropy output distributions (Szegedy et al., 2016). LSR can reduce overfitting by preventing a network from assigning the full probability to each training example during training, replaces the 0 and 1 targets for a classifier with smoothed values like 0.1 or 0.9.

For a training sample x with the original ground-truth label distribution $q(k|x)$, we replace $q(k|x)$ with

$$q(k|x) = (1 - \epsilon)q(k|x) + \epsilon u(k) \tag{15}$$

where $u(k)$ is the prior distribution over labels , and ϵ is the smoothing parameter. In this paper, we set the prior label distribution to be uniform $u(k) = 1/C$.

LSR is equivalent to the KL divergence between the prior label distribution $u(k)$ and the network's predicted distribution p_θ. Formally, LSR term is defined as:

$$\mathcal{L}_{lsr} = -D_{KL}(u(k) \| p_\theta) \tag{16}$$

The objective function (loss function) to be optimized is the cross-entropy loss with $\mathcal{L}_{lsr}$ and $\mathcal{L}_2$ regularization, which is defined as:

$$\mathcal{L}(\theta) = -\sum_{i=1}^{C} \hat{y}^c log(y^c) + \mathcal{L}_{lsr} + \lambda \sum_{\theta \in \Theta} \theta^2 \tag{17}$$

where $\hat{y} \in \mathbb{R}^C$ is the ground truth represented as a one-hot vector, y is the predicted sentiment distribution vector given by the output layer, λ is the coefficient for $\mathcal{L}_2$ regularization term, and Θ is the parameter set.

4 Experiments

4.1 Datasets and Experimental Settings

We conduct experiments on three datasets: SemEval 2014 Task 4 [2] (Pontiki et al., 2014) dataset composed of *Restaurant* reviews and *Laptop* reviews, and ACL 14 *Twitter* dataset gathered by Dong et al. (2014). These datasets are labeled with three sentiment polarities: *positive*, *neutral* and *negative*. Table 1 shows the number of training and test instances in each category.

Word embeddings in AEN-GloVe do not get updated in the learning process, but we fine-tune pre-trained BERT [3] in AEN-BERT. Embedding dimension d_{dim} is 300 for GloVe and is 768 for pre-trained BERT. Dimension of hidden states d_{hid} is set to 300. The weights of our model are initialized with Glorot initialization (Glorot and Bengio, 2010). During training, we set label smoothing parameter ϵ to 0.2 (Szegedy et al., 2016), the coefficient λ of $\mathcal{L}_2$ regularization item is 10^{-5} and dropout rate is 0.1. Adam optimizer (Kingma and Ba, 2014) is applied to update all the parameters. We adopt the *Accuracy* and *Macro-F1* metrics to evaluate the performance of the model.

[2]The detailed introduction of this task can be found at `http://alt.qcri.org/semeval2014/task4`.

[3]We use uncased BERT-base from `https://github.com/google-research/bert`.

Table 1: Statistics of the datasets.

Dataset	Positive		Neural		Negative	
	Train	Test	Train	Test	Train	Test
Twitter	1561	173	3127	346	1560	173
Restaurant	2164	728	637	196	807	196
Laptop	994	341	464	169	870	128

4.2 Model Comparisons

In order to comprehensively evaluate and analysis the performance of AEN-GloVe, we list 7 baseline models and design 4 ablations of AEN-GloVe. We also design a basic BERT-based model to evaluate the performance of AEN-BERT.

Non-RNN based baselines:

- **Feature-based SVM** (Kiritchenko et al., 2014) is a traditional support vector machine based model with extensive feature engineering.
- **Rec-NN** (Dong et al., 2014) firstly uses rules to transform the dependency tree and put the opinion target at the root, and then learns the sentence representation toward target via semantic composition using Recursive NNs.
- **MemNet** (Tang et al., 2016b) uses multi-hops of attention layers on the context word embeddings for sentence representation to explicitly captures the importance of each context word.

RNN based baselines:

- **TD-LSTM** (Tang et al., 2016a) extends LSTM by using two LSTM networks to model the left context with target and the right context with target respectively. The left and right target-dependent representations are concatenated for predicting the sentiment polarity of the target.
- **ATAE-LSTM** (Wang et al., 2016) strengthens the effect of target embeddings, which appends the target embeddings with each word embeddings and use LSTM with attention to get the final representation for classification.
- **IAN** (Ma et al., 2017) learns the representations of the target and context with two LSTMs and attentions interactively, which generates the representations for targets and contexts with respect to each other.
- **RAM** (Chen et al., 2017) strengthens MemNet by representing memory with bidirectional LSTM and using a gated recurrent unit network to combine the multiple attention outputs for sentence representation.

AEN-GloVe ablations:

- **AEN-GloVe w/o PCT** ablates PCT module.
- **AEN-GloVe w/o MHA** ablates MHA module.
- **AEN-GloVe w/o LSR** ablates label smoothing regularization.
- **AEN-GloVe-BiLSTM** replaces the attentional encoder layer with two bidirectional LSTM.

Basic BERT-based model:

- **BERT-SPC** feeds sequence "[CLS] + context + [SEP] + target + [SEP]" into the basic BERT model for sentence pair classification task.

4.3 Main Results

Table 2 shows the performance comparison of AEN with other models. BERT-SPC and AEN-BERT obtain substantial accuracy improvements, which shows the power of pre-trained BERT on small-data task. The overall performance of AEN-BERT is better than BERT-SPC, which suggests that it is important to design a downstream network customized to a specific task. As the prior knowledge in the pre-trained BERT is not specific to any particular domain, further fine-tuning on the specific task is necessary for releasing the true power of BERT.

The overall performance of TD-LSTM is not good since it only makes a rough treatment of the target words. ATAE-LSTM, IAN and RAM are attention based models, they stably exceed the TD-LSTM method on *Restaurant* and *Laptop* datasets. RAM is better than other RNN based models, but it does not perform well on *Twitter* dataset, which might because bidirectional LSTM is not good at modeling small and ungrammatical text.

Feature-based SVM is still a competitive baseline, but relying on manually-designed features. Rec-NN gets the worst performances among all neural network baselines as dependency parsing is not guaranteed to work well on ungrammatical short texts such as tweets and comments. Like AEN, MemNet also eschews recurrence, but its overall performance is not good since it does not model the hidden semantic of embeddings, and the result of the last attention is essentially a linear combination of word embeddings.

4.4 Model Analysis

As shown in Table 2, the performances of AEN-GloVe ablations are incomparable with AEN-

Table 2: Main results. The results of baseline models are retrieved from published papers. "-" means not reported. Top 3 scores are in **bold**.

	Models	Twitter		Restaurant		Laptop	
		Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
RNN baselines	TD-LSTM	0.7080	0.6900	0.7563	-	0.6813	-
	ATAE-LSTM	-	-	0.7720	-	0.6870	-
	IAN	-	-	0.7860	-	0.7210	-
	RAM	0.6936	0.6730	0.8023	0.7080	**0.7449**	**0.7135**
Non-RNN baselines	Feature-based SVM	0.6340	0.6330	0.8016	-	0.7049	-
	Rec-NN	0.6630	0.6590	-	-	-	-
	MemNet	0.6850	0.6691	0.7816	0.6583	0.7033	0.6409
AEN-GloVe ablations	AEN-GloVe w/o PCT	0.7066	0.6907	0.8017	0.7050	0.7272	0.6750
	AEN-GloVe w/o MHA	0.7124	0.6953	0.7919	0.7028	0.7178	0.6650
	AEN-GloVe w/o LSR	0.7080	0.6920	0.8000	0.7108	0.7288	0.6869
	AEN-GloVe-BiLSTM	0.7210	**0.7042**	0.7973	0.7037	0.7312	0.6980
Ours	AEN-GloVe	**0.7283**	0.6981	**0.8098**	**0.7214**	0.7351	0.6904
	BERT-SPC	**0.7355**	**0.7214**	**0.8446**	**0.7698**	**0.7899**	**0.7503**
	AEN-BERT	**0.7471**	**0.7313**	**0.8312**	**0.7376**	**0.7993**	**0.7631**

GloVe in both accuracy and macro-F1 measure. This result shows that all of these discarded components are crucial for a good performance. Comparing the results of AEN-GloVe and AEN-GloVe w/o LSR, we observe that the accuracy of AEN-GloVe w/o LSR drops significantly on all three datasets. We could attribute this phenomenon to the unreliability of the training samples with *neutral* sentiment. The overall performance of AEN-GloVe and AEN-GloVe-BiLSTM is relatively close, AEN-GloVe performs better on the *Restaurant* dataset. More importantly, AEN-GloVe has fewer parameters and is easier to parallelize.

To figure out whether the proposed AEN-GloVe is a lightweight alternative of recurrent models, we study the model size of each model on the *Restaurant* dataset. Statistical results are reported in Table 3. We implement all the compared models base on the same source code infrastructure, use the same hyperparameters, and run them on the same GPU [4].

RNN-based and BERT-based models indeed have larger model size. ATAE-LSTM, IAN, RAM, and AEN-GloVe-BiLSTM are all attention based RNN models, memory optimization for these models will be more difficult as the encoded hidden states must be kept simultaneously in memory in order to perform attention mechanisms. MemNet has the lowest model size as it only has one shared attention layer and two linear layers, it does not calculate hidden states of word embeddings. AEN-GloVe's lightweight level ranks second, since it takes some more parameters than MemNet in modeling hidden states of sequences. As a comparison, the model size of AEN-GloVe-BiLSTM is more than twice that of AEN-GloVe, but does not bring any performance improvements.

[4]NVIDIA GTX 1080ti.

Table 3: Model sizes. Memory footprints are evaluated on the Restaurant dataset. Lowest 2 are in **bold**.

Models	Model size	
	Params $\times 10^6$	Memory (MB)
TD-LSTM	1.44	12.41
ATAE-LSTM	2.53	16.61
IAN	2.16	15.30
RAM	6.13	31.18
MemNet	**0.36**	**7.82**
AEN-BERT	112.93	451.84
AEN-GloVe-BiLSTM	3.97	22.52
AEN-GloVe	**1.16**	**11.04**

5 Conclusion

In this work, we propose an attentional encoder network for the targeted sentiment classification task. which employs attention based encoders for the modeling between context and target. We raise the the label unreliability issue add a label smoothing regularization to encourage the model to be less confident with fuzzy labels. We also apply pre-trained BERT to this task and obtain new state-of-the-art results. Experiments and analysis demonstrate the effectiveness and lightweight of the proposed model.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 452–461.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Xiaowen Ding, Bing Liu, and Philip S Yu. 2008. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 international conference on web search and data mining*, pages 231–240. ACM.

Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. 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)*, volume 2, pages 49–54.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 151–160. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. 2014. Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 437–442.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018. Transformation networks for target-oriented sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 946–956.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4068–4074. AAAI Press.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35.

Delip Rao and Deepak Ravichandran. 2009. Semi-supervised polarity lexicon induction. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 675–682. Association for Computational Linguistics.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective lstms for target-dependent sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3298–3307.

Duyu Tang, Bing Qin, and Ting Liu. 2016b. Aspect level sentiment classification with deep memory network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 214–224.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Yequan Wang, Minlie Huang, Li Zhao, et al. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615.

Paul J Werbos. 1990. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560.