

METNet: A Mutual Enhanced Transformation Network for Aspect-based Sentiment Analysis

Bin Jiang^{1*}, Jing Hou¹, Wanyue Zhou¹, Chao Yang¹, Shihan Wang² and Liang Pang³

¹College of Computer Science and Electronic Engineering, Hunan University, China

²Intelligent Systems, Utrecht University, The Netherlands

³CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology, Chinese Academy of Sciences, China

{jiangbin, houjing, zhouwanyue, yangchaoedu}@hnu.edu.cn
s.wang2@uu.nl, pangliang@ict.ac.cn

Abstract

Aspect-based sentiment analysis (ABSA) aims to determine the sentiment polarity of each specific aspect in a given sentence. Existing researches have realized the importance of the aspect for the ABSA task and have derived many interactive learning methods that model context based on specific aspect. However, current interaction mechanisms are ill-equipped to learn complex sentences with multiple aspects, and these methods underestimate the representation learning of the aspect. In order to solve the two problems, we propose a mutual enhanced transformation network (METNet) for the ABSA task. First, the aspect enhancement module in METNet improves the representation learning of the aspect with contextual semantic features, which gives the aspect more abundant information. Second, METNet designs and implements a hierarchical structure, which enhances the representations of aspect and context iteratively. Experimental results on SemEval 2014 Datasets demonstrate the effectiveness of METNet, and we further prove that METNet is outstanding in multi-aspect scenarios.

1 Introduction

Aspect-based sentiment analysis (ABSA) aims to determine the orientation of sentiment expressed on each aspect in a sentence (Liu, 2012). For instance, in the sentence “Although the **service** is not that great, I still like the **food**”, the user mentions two aspects “service” and “food” and expresses negative sentiment and positive sentiment respectively. The ABSA task consists of two subtasks including aspect extraction and aspect sentiment classification (Zhang et al., 2018). In this paper, we assume that the aspects are given and only focus on the aspect sentiment classification task. In ABSA task, multiple aspects could appear in one sentence. When predicting the sentiment of current aspect, words related to other aspects with different sentiment tendencies maybe become noises. Therefore, how to effectively model the semantic relationship between given aspect and context words is an important challenge.

The traditional methods mainly rely on manually designed features, which is labor-intensive, and this representation method almost reaches its performance bottleneck (Ma et al., 2017). Boosted by the recent development of deep learning techniques, some studies utilize attention mechanism to solve the above problems well, and many neural attention models have been proposed (Wang et al., 2016; Tang et al., 2016b; Chen et al., 2017; Ma et al., 2017). In these works, the model generally obtains the aspect representation first and then applies attention mechanism to extract the context features related to the given aspect for sentiment prediction. However, the attention mechanism has some drawbacks. When a sentence contains multiple aspects with different sentiment tendencies, opinion modifiers of other aspects are noise information for the current aspect. But, it is hard for the attention mechanism to differentiate opinion modifiers of multiple aspects, which directly affects the final prediction result. For example, in the sentence “I like coming back to **Mac OS** but this laptop is lacking in **speaker quality** compared to my \$400 old HP laptop”, the attention mechanism should pay more attention to the opinion word “like” with positive sentiment tendency for aspect “Mac OS”. However, the attention mechanism tends to involve irrelevant opinion words, such as “lacking” with negative sentiment tendency, which interferes with the

*Corresponding author: Bin Jiang.

This work is licensed under a Creative Commons Attribution 4.0 International License. License details: <http://creativecommons.org/licenses/by/4.0/>.

sentiment prediction of aspect “Mac OS”. To this end, some researchers put forward many works (Li et al., 2018; Ran et al., 2019) to overcome the shortcomings inherent in attention mechanism for the ABSA task. However, most of these works are devoted to designing complex neural networks to improve the representation learning of the context. Few works focus on how to improve the representation learning of the aspect. In recent work, Yang et al. (2019) learns the effective representations of aspect and context alternately with an iteration mechanism and finally obtains more accurate prediction results.

In order to make the aspect play a better role in the ABSA task, we propose the mutual enhanced transformation network (METNet). METNet improves the representation learning of the aspect and extracts more effective context features based on enhanced aspect representation. Inspired by TNet (Li et al., 2018) and Coattention-MemNet (Yang et al., 2019), METNet utilizes the hierarchical structure to learn the representations of aspect and context iteratively, and we name each computational layer as Bidirectional Enhancement Transformation (BET) layer. Each BET layer has three parts, including a bidirectional LSTM, an aspect enhancement module, and the aspect-specific transformation (AST) units. Specifically, BET learns the context dependency of sentence by a bidirectional LSTM first. Then, the aspect enhancement module utilizes the extracted context features to improve the aspect representation. After that, the AST units fuse the aspect information into each context word to obtain a sentence-level context representation. Contexts and aspects passing through multiple BET layers are eventually fed into a sentiment feature extractor. Since GCAE (Xue and Li, 2018) adds aspect information when extracting sentiment features, which further strengthens the connection between aspect and context. We replace the feature extractor from CNN to GCAE. Moreover, in order to help GCAE extract sentiment features more accurately, we utilize relative position information to scale the input of GCAE.

The main contributions of this work are summarized as follows: (1) We propose an aspect enhancement module which utilizes the extracted context features to enhance aspect representation. The enhanced aspect representation is utilized to obtain the more effective context representation. (2) Based on the aspect enhancement module, we propose the mutual enhanced transformation network (METNet) for aspect-based sentiment analysis which learns the representations of aspect and context alternately and iteratively. Experimental results confirm the effectiveness of METNet, and METNet is outstanding in multi-aspect scenarios.

2 Related Works

The traditional methods (Boiy and Moens, 2009; Kiritchenko et al., 2014) for the ABSA task mainly utilize manually designed features such as sentiment lexicon, n-grams, and dependency information, which is labor-intensive. Also, the quality of the features directly affects the classification accuracy of methods.

With the development of deep learning on natural language processing tasks, many neural network models (Tang et al., 2016a; Wang et al., 2016; Ma et al., 2017; Li et al., 2018; Yang et al., 2019; Ran et al., 2019) have been proposed. These methods automatically learn sentiment features of sentence and achieve good results. Tang et al. (2016a) adopted two LSTMs to model the two clauses of given sentence and then integrated the coding features from the two LSTMs for prediction.

Recently, because attention mechanism can clearly capture the semantic relationship between given aspect and words in sentence, attention-based neural models have attracted growing interest. Wang et al. (2016) proposed an attention-based LSTM network. Tang et al. (2016b) adopted an end-to-end memory network architecture where each computational layer was based on the attention mechanism. Li et al. (2018) utilized the attention mechanism to generate different aspect representations based on individual context words and then extracted context features based on tailor-made aspect representations.

Existing methods pay little attention to how to improve the representation learning of the aspect. Inspired by TNet (Li et al., 2018) and Coattention-MemNet (Yang et al., 2019), we propose the mutual enhanced transformation network (METNet) for aspect-based sentiment analysis. The main differences between our method and existing methods are as follows: (1) METNet improves the representation learning of the aspect. (2) We propose the Bidirectional Enhancement Transformation (BET) component which can be repeated by the hierarchical structure to obtain more effective representations of aspect and context.

3 Model Overview

In this section, we describe our proposed METNet model which is shown in Figure 1. The METNet model can be roughly divided into three parts, namely the BERT Layer, the Bidirectional Enhancement Transformation (BET), and the Convolutional Feature Extractor.

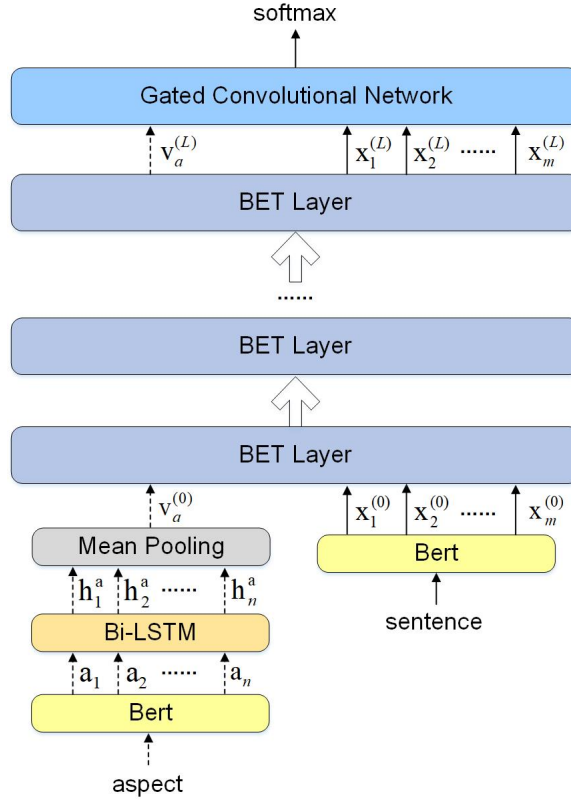


Figure 1: The overall architecture of METNet, where the dotted arrows represent the transformation of aspect, and the solid arrows represent the transformation of sentence.

3.1 BERT Layer

BERT has been successfully applied to process various NLP tasks, such as Question Answer and Dialog System. The BERT layer uses the pre-trained BERT to generate word representations of sequence. Suppose that a sentence consists of m words and an aspect contains n words. Then, we can obtain the vector representation $x^{(0)} = \{x_1^{(0)}, x_2^{(0)}, \dots, x_m^{(0)}\} \in \mathbb{R}^{m \times d}$ of the sentence and the vector representation $a = \{a_1, a_2, \dots, a_n\} \in \mathbb{R}^{n \times d}$ of the aspect by the BERT layer, where d denotes the dimension of the BERT output layer.

3.2 Bidirectional Enhancement Transformation (BET)

The bidirectional enhancement transformation (BET) layer in Figure 1 is introduced in this section, while its details are shown in Figure 2. Each BET layer contains three parts, namely a Bi-directional LSTM Layer, an Aspect Enhancement Module, and a set of Aspect-Specific Transformation (AST) Units. The BiLSTM layer first generates the contextualized word representations based on the input. Then, the aspect enhancement module uses the word representations to enhance the aspect representation. Finally, the AST units generate the aspect-specific word representations based on the contextualized word representations and the enhanced aspect representation. Details are described below.

Bi-directional LSTM Layer: As mentioned earlier, we first use BERT to encode sentence. BERT’s model architecture is a multi-layer bidirectional Transformer encoder (Devlin et al., 2019). Yan et al. (2019) analyzed the shortcomings of the Transformer, that is, the Transformer weakened the directional information and relative position information of the text. However, the relative position information of

the text is important for the ABSA task. Therefore, we set up a BiLSTM after BERT to learn the context dependency of the text.

As shown in Figure 1, we repeat the BET layer by the hierarchical structure. The input of the BiLSTM layer in the bottom BET layer is the context representation outputted by BERT, and the input of the BiLSTM layer in the next BET layer is from the outputs of AST units in the previous BET layer.

We formulate the word representation outputted by the BiLSTM as $h^{(i)} = \{h_1^{(i)}, h_2^{(i)}, \dots, h_m^{(i)}\}$. The forward LSTM outputs a set of hidden states $\overrightarrow{h}^{(i)} = \{\overrightarrow{h}_1^{(i)}, \overrightarrow{h}_2^{(i)}, \dots, \overrightarrow{h}_m^{(i)}\} \in \mathbb{R}^{m \times d_h}$, where d_h denotes the number of hidden units. Similarly, the backward LSTM also outputs a set of hidden states $\overleftarrow{h}^{(i)} = \{\overleftarrow{h}_1^{(i)}, \overleftarrow{h}_2^{(i)}, \dots, \overleftarrow{h}_m^{(i)}\} \in \mathbb{R}^{m \times d_h}$. Finally, we connect two hidden state lists to obtain the word representation $h^{(i)} = \{h_1^{(i)}, h_2^{(i)}, \dots, h_m^{(i)}\} \in \mathbb{R}^{m \times 2d_h}$, where $h_j^{(i)} = [\overrightarrow{h}_j^{(i)}; \overleftarrow{h}_j^{(i)}] \in \mathbb{R}^{2d_h}$.

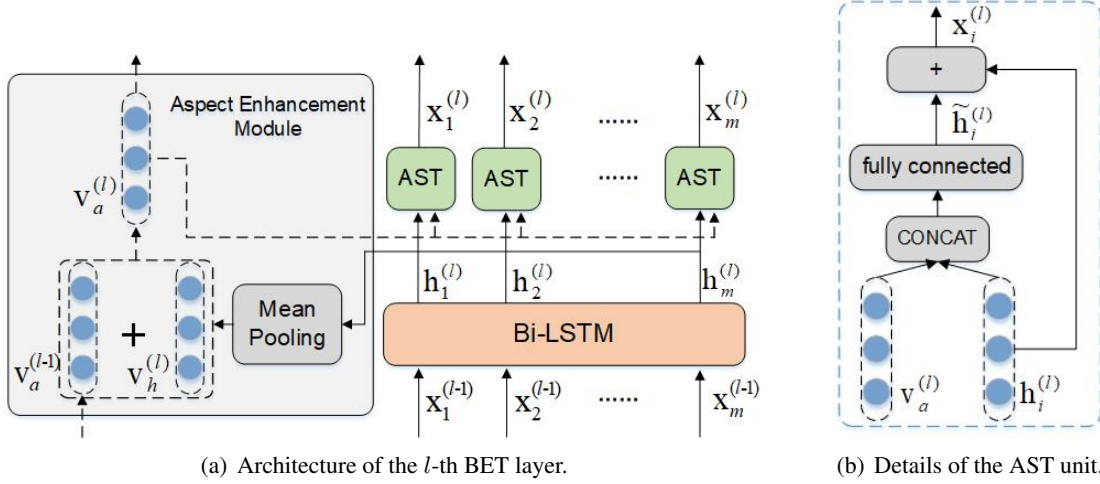


Figure 2: Architecture of the l -th BET layer and details of the AST unit.

Aspect Enhancement Module: Before introducing this module in detail, we first introduce how to obtain the initial aspect representation. Specifically, we feed the aspect vector $a = \{a_1, a_2, \dots, a_n\} \in \mathbb{R}^{n \times d}$ outputted by BERT into another BiLSTM and then apply the average pooling method to the obtained hidden state vectors $h^a = \{h_1^a, h_2^a, \dots, h_n^a\} \in \mathbb{R}^{n \times 2d_h}$. Finally, we can get the initial aspect representation $v_a^{(0)} \in \mathbb{R}^{2d_h}$, as shown in Figure 1.

Take the bottom BET layer as an example to introduce the aspect enhancement module in detail. First, based on the contextualized context representation $h^{(1)}$ outputted by BiLSTM, we use the average pooling layer to obtain a vector $v_h^{(1)} \in \mathbb{R}^{2d_h}$, namely the contextual vector. Then, we use a basic feature fusion method, the point-wise addition, to fuse contextual vector into the initial aspect representation, which can be formulated as $v_a^{(1)} = v_a^{(0)} + v_h^{(1)} \in \mathbb{R}^{2d_h}$. This is an enhancement operation on the aspect. By analogy, the final aspect representation is $v_a^{(L)} = v_a^{(L-1)} + v_h^{(L)} \in \mathbb{R}^{2d_h}$, and we unfold this formula as follows:

$$v_a^{(L)} = v_a^{(0)} + v_h^{(1)} + v_h^{(2)} + \dots + v_h^{(L)} \quad (1)$$

where $v_h^{(i)} \in \mathbb{R}^{2d_h}$, $i \in [1, L]$ represents the contextual vector in the i -th BET layer. From Eq.1, we can see that the aspect representation is gradually strengthened by different contextual vectors in multiple BET layers.

As shown in Figure 2, the aspect vector $v_a^{(l)}$, $l \in [1, L-1]$ has two directions, one is AST units in the same BET layer, and another is the aspect enhancement module in the next BET layer.

Aspect-Specific Transformation (AST) Unit: As mentioned earlier, one direction of the aspect vector $v_a^{(l)}$ is AST units in the same BET layer. The role of AST units is to generate aspect-specific word representations. The AST unit uses a structure similar to the CPT module in TNet proposed by Li et al. (2018).

The AST unit takes the aspect vector $v_a^{(l)}$ and the word representation $h_i^{(l)}$ as inputs, where $h_i^{(l)}$ is the i -th word-level representation outputted by the BiLSTM layer. $v_a^{(l)}$ is the enhanced aspect vector. Specifically, the concatenation of $v_a^{(l)}$ and $h_i^{(l)}$ is fed into a fully connected layer to obtain the i -th aspect-specific word representation $\tilde{h}_i^{(l)}$:

$$\tilde{h}_i^{(l)} = g(\widetilde{W}[h_i^{(l)} : v_a^{(l)}] + \widetilde{b}) \quad (2)$$

where $g(*)$ is a non-linear activation function, and “:” denotes vector concatenation. \widetilde{W} and \widetilde{b} are weight matrix and bias respectively.

There is an information protection mechanism to ensure that the context dependency information captured from BiLSTM will not be lost. This information protection mechanism strengthens the transmission and use of features and can be formulated as:

$$\mathbf{x}_i^{(l)} = h_i^{(l)} + \tilde{h}_i^{(l)} \quad (3)$$

where $\mathbf{x}_i^{(l)}$ is the output of the AST unit.

3.3 Convolutional Feature Extractor

In this subsection, we will introduce a feature extractor for extracting sentiment features which is a gated convolutional network (GCAE) proposed by Xue and Li (2018). GCAE differs from vanilla CNN. The ReLU gate in GCAE receives the aspect information to control the propagation of sentiment features, which further enhances the connection between aspect and context.

Also, Chen et al. (2017) introduced position information, which makes the model pay more attention to sentiment modifiers that are closer to the current aspect, thereby improving the classification accuracy of the model in multi-aspect scenarios. Inspired by this work, a variable p_i is introduced in our model to measure the relative position information between the i -th word in the context and the current aspect. Specifically, the calculation of p_i is as follows¹:

$$p_i = \begin{cases} 1 - \frac{(k+n-i)}{C} & i < k + n \\ 1 - \frac{i-k}{C} & k + n \leq i \leq m \\ 0 & i > m \end{cases} \quad (4)$$

where k is the index of the first aspect word, C is a pre-specified constant, and n is the length of the aspect. Then, we multiply p_i as the weight by the word representation outputted by the i -th AST unit in the L -th BET layer:

$$\mathbf{x}_i = p_i * \mathbf{x}_i^{(L)} \quad (5)$$

Then, we feed $X = \{x_1, x_2, \dots, x_m\}$ and the final aspect vector $v_a^{(L)}$ into the above gated convolutional network to generate a feature map c :

$$a_i = \text{relu}(W_a X_{i:i+k-1} + V_a v_a^{(L)} + b_a) \quad (6)$$

$$s_i = \text{tanh}(W_s X_{i:i+k-1} + b_s) \quad (7)$$

$$c_i = s_i \times a_i \quad (8)$$

where k is the kernel size, and W_a , V_a , b_a , W_s , and b_s are learnable parameters. \times denotes element-wise multiplication. Then, we apply max pooling (Kim, 2014) and obtain the sentence representation z by employing s kernels:

$$z = \{\max(c_1), \dots, \max(c_s)\} \quad (9)$$

Finally, we pass z to a fully connected layer for sentiment prediction:

¹As we perform sentence padding, it is possible that the index i is larger than the actual length m of the sentence.

$$\hat{y} = \text{softmax}(W_f z + b_f) \quad (10)$$

where W_f and b_f are learnable parameters.

3.4 Model training

METNet can be trained in an end-to-end manner within a supervised learning framework to optimize all the parameters notated as Θ . Cross entropy with L_2 regularization is used as the loss function, which is defined as:

$$L = - \sum_{i=1}^O y_i \log(\hat{y}_i) + \lambda \left(\sum_{\theta \in \Theta} \theta^2 \right) \quad (11)$$

where y_i denotes the ground truth, and \hat{y}_i is the estimated probability for each sentiment. O stands for the number of sentiment polarities. λ is the coefficient for L_2 regularization.

4 Experiments

4.1 Experimental Setup

Datasets: We evaluate the proposed METNet on three benchmark datasets: Restaurant and Laptop datasets are from SemEval 2014 (Pontiki et al., 2014); Twitter dataset is a collection of tweets, introduced by Dong et al. (2014). In these three datasets, each review corresponds to an aspect and a sentiment polarity. We consider three types of sentiment polarity, including positive, negative, and neutral. The detailed statistics of datasets are shown in Table 1.

Dasesets		Positive	Negative	Neutral	Total
Laptop	Train	994	870	464	2328
	Test	341	128	169	638
Restaurant	Train	2164	807	637	3608
	Test	728	196	196	1120
Twitter	Train	1561	1560	3127	6248
	Test	173	173	346	692

Table 1: Statistics of datasets.

Parameters Setting: We use the Bert-base-uncased pre-trained model which contains 12 layers and its hidden layer dimension is 768. All the weight matrices are given the initial value by sampling from the uniform distribution $U(-0.1, 0.1)$. We adopt the dropout strategy after BERT, and the dropout rate is set to 0.5. The coefficient for L_2 regularization is set to 10^{-4} . The number of hidden units of BiLSTM d_h , the constant C , and the size k and number s of the convolution kernel are set to 384, 40, 3, and 50, respectively. We train the model with the Adadelta (Zeiler, 2012) optimizer and set learning rate as 1.

4.2 Compared Methods

To justify the effectiveness of our METNet, we compared it with the following methods.

SVM (Kiritchenko et al., 2014): It is a traditional support vector machine based model with extensive feature engineering.

AE-LSTM (Wang et al., 2016): AE-LSTM is an attention-based LSTM network, which uses the attention mechanism to calculate the correlation between aspect and words in sentence.

ATAE-LSTM (Wang et al., 2016): ATAE-LSTM is an extension of AE-LSTM. Considering the importance of aspect for sentiment prediction, ATAE-LSTM adds aspect embedding to the input of the model.

IAN (Ma et al., 2017): IAN interactively learns attentions in the context and aspect and generates the representations for context and aspect separately.

MemNet (Tang et al., 2016b): MemNet contains multiple computational layers that share parameters. Each layer is a model based on content attention and location attention and assigns weights to each context word to obtain a sentence representation.

Cabasc (Liu et al., 2018): Cabasc uses two attention enhancement mechanisms to flexibly model the word order information, the aspect information, and the correlation between the word and the aspect.

TNet-LF (Li et al., 2018): TNet-LF proposes to dynamically compute the importance of each aspect word based on each context word rather than the whole sentence.

TNet-ATT(+AS) (Tang et al., 2019): TNet-ATT(+AS) is developed based on TNet-LF. TNet-ATT(+AS) proposes an algorithm that can automatically mine attention supervision information, thereby improving the model’s insufficient learning of low-frequency words with sentiment polarity.

IARM (Majumder et al., 2018): IARM generates independent aspect-aware sentence representations for all aspects in a sentence to help predict the sentiment polarity of current aspect.

HGMN (Ran et al., 2019): HGMN distills out aspect-specific effective text spans in sentence instead of only the aggregated contextual representation based on attention score.

Coattention-MemNet (Yang et al., 2019): Coattention-MemNet learns the key features from the aspect and context alternately with an iteration mechanism.

Models	Laptop		Restaurant		Twitter	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
SVM	70.49	-	80.16	-	63.40*	63.30*
AE-LSTM	68.90	-	76.60	-	-	-
ATAE-LSTM	68.81 [#]	63.11 [#]	77.68 [#]	64.89 [#]	67.49 [#]	65.19 [#]
MemNet	70.33 ^{**}	64.09 ^{**}	78.16 ^{**}	65.83 ^{**}	68.50 ^{**}	66.91 ^{**}
IAN	72.57 [#]	66.73 [#]	78.48 [#]	67.55 [#]	70.81 [#]	67.86 [#]
Cabasc	75.07	-	80.89	-	71.53	-
TNet-LF	75.08 [#]	69.78 [#]	80.18 [#]	70.06 [#]	74.68	73.36
TNet-ATT(+AS)	77.62	73.84	81.53	72.90	78.61	77.72
IARM	73.80	-	80.00	-	-	-
HGMN	76.67	72.22	82.33	73.34	73.70	72.89
Coattention-MemNet	72.90	-	79.70	-	70.50	-
METNet w/o aspect enhancement module	78.06	73.88	81.34	71.50	73.12	71.25
METNet	78.37	74.93	82.50	73.92	74.42	73.00

Table 2: Experimental results (%). The best results are in bold. The results with symbol “#” are reproduced under the same conditions with the original paper. Those starred (*) results are from Dong et al. (2014) and starred (**) results are from Li et al. (2018). Other results are retrieved from the original papers.

Table 2 statistics the performance of each model on three datasets. The main evaluation metrics are Accuracy and Macro-averaged F1-score.

4.3 Experimental Results and Analysis

Analysis of METNet: We compare the classification accuracy of METNet with all baselines, and the main results are shown in Table 2. As we can see from the results, METNet outperforms all baselines on Laptop and Restaurant datasets. TNet-ATT(+AS) achieves the best result on Laptop dataset among all baselines due to its progressive self-supervised attention learning approach. HGMN achieves the best result on Restaurant dataset among all baselines due to its hierarchical gate mechanism. METNet achieves 0.75% and 0.17% accuracy improvements on Laptop and Restaurant datasets compared with TNet-ATT(+AS) and HGMN, which indicates the effectiveness of METNet. METNet did not perform well on Twitter dataset. The reason may be that METNet is not suitable for Twitter dataset that is composed of single-aspect sentences.

In addition, in order to highlight the advantages of METNet in multi-aspect scenarios, we conduct further experiments and introduce the relevant experimental results of IARM from the original paper. Specifically, we first delete the single-aspect samples in the Laptop testset and Restaurant testset and mark the new datasets as Laptop* and Restaurant*. Then, we apply the trained TNet-LF and METNet to Laptop* and Restaurant* datasets and present the results in Table 3. We find that METNet achieves significant accuracy improvements, which indicates the effectiveness of METNet in multi-aspect scenarios.

Models	Laptop*		Restaurant*	
	Accuracy	Macro-F1	Accuracy	Macro-F1
TNet-LF	74.80	67.34	80.31	69.35
IARM	74.10 [#]	-	80.48 [#]	-
METNet w/o aspect enhancement module	76.38	72.50	81.03	69.97
METNet	77.95	73.80	82.59	73.92

Table 3: Experimental results (%) on Laptop* and Restaurant* datasets. The best results are in bold. The results with symbol “#” are retrieved from the original paper.

Effects of Aspect Enhancement Module: We validate that the aspect enhancement module is effective for the ABSA task by comparing METNet and METNet w/o aspect enhancement module. Note that METNet w/o sth represents METNet with sth removed, that is, METNet w/o aspect enhancement module is the version where METNet removed the aspect enhancement module. In order to prove the above point, we conduct ablation experiments. The results are shown in Table 2. Compared with METNet w/o aspect enhancement module, METNet achieves 0.31%, 1.16%, and 1.30% accuracy improvements on Laptop, Restaurant, and Twitter datasets respectively, which proves the effectiveness of the aspect enhancement module.

On the basis of the above, we also conduct a set of experiments to examine the importance of the aspect enhancement module in multi-aspect scenarios. We apply the trained METNet w/o aspect enhancement module to Laptop* and Restaurant* datasets and then record and analyze the experimental results. In Table 3, METNet achieves 1.57% and 1.56% accuracy improvements on Laptop* and Restaurant* datasets compared with METNet w/o aspect enhancement module, which effectively proves the importance of aspect information in multi-aspect scenarios. We also observe in Table 3 that METNet w/o aspect enhancement module outperforms TNet-LF and IARM, which may benefit from the BET layer we designed and the application of GCAE.

Number of layers	Laptop		Restaurant		Twitter	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
1	78.21	74.29	82.23	73.37	73.84	72.30
2	78.37	74.93	82.50	73.92	74.42	73.00
3	77.90	74.10	82.23	73.25	74.28	72.62
4	77.27	72.47	81.43	71.47	74.13	72.51

Table 4: Impact of BET Layer Number.

Impact of BET Layer Number: As our METNet involves multiple BET layers, we investigate the effect of the layer number LN. Specifically, we conduct experiments on Laptop, Restaurant, and Twitter datasets and vary LN from 1 to 4. In Table 4, we can see that METNet achieves the best results when LN is 2. We also have tried to set the LN larger, and the classification accuracy basically becomes worse, which is probably because more parameters increase the training difficulty.

4.4 Case Study

We pick some test examples from the testset to evaluate the performance of METNet further and present several example sentences in Table 4. In the sentence (5) and (6), the sentiment of aspects is determined by related opinion words, and the proposed METNet can make correct predictions even without using the attention mechanism. Moreover, METNet is better at handling long and complex sentences with

multiple aspects compared with ATAE-LSTM and TNet. For example, sentence (1), (6), (7) are typical long reviews and involve multiple aspects. In this statement, METNet makes correct predictions on all aspects. Because compared with ATAE-LSTM and TNet, METNet has the aspect enhancement module which gives aspect with rich semantic information and enhances the connection between aspect and context. In addition, we find METNet, ATAE-LSTM and TNet perform poorly on sentences (2), (3). In this statement, the difficulty of prediction comes from the comparison between aspects. It is the inference based on implicit semantics, which is still quite challenging for neural network models.

Sample Sentences	ATAE-LSTM	TNet	METNet
(1) Would you ever believe that when you complain about over an hour <u>wait</u> _N , when they tell you it will be 20-30 minutes, the <u>manager</u> _N tells the <u>bartender</u> _O to spill the <u>drinks</u> _O you just paid for?	N, N, N , O	N, N, N , O	N, N, O, O
(2) <u>New hamburger with special sauce</u> _P is ok - at least better than <u>big mac</u> _N !	O , P	N , N	P , P
(3) <u>Price</u> _N was higher when purchased on mac when compared to <u>price</u> _P showing on pc when I bought this product.	N, N	N, N	N, N
(4) Great <u>food</u> _P but the <u>service</u> _N was dreadful!	P , P	P, N	P, N
(5) They really provide a relaxing, laid-back <u>atmosphere</u> _P .	P	P	P
(6) Not only did they have amazing, <u>sandwiches</u> _P , <u>soup</u> _P , <u>pizza</u> _P etc, but their <u>homemade sorbets</u> _P are out of this world!	P, P, P, P	N , P, P, P	P, P, P, P
(7) This is one great place to eat <u>pizza</u> _P more out but not a good place for <u>take-out pizza</u> _N .	P , P	P, N	P, N

Table 5: Some test examples and predictions of ATAE-LSTM, TNet, and METNet. The aspects are underlined with the true labels given as subscripts. The notations P, N, and O represent positive, negative, and neutral respectively. Incorrect predictions are highlighted in red color.

5 Conclusion

In this paper, we present an end-to-end solution, a mutual enhanced transformation network (METNet), to solve the issue of insufficient aspect representation learning in ABSA task. First, we propose Bidirectional Enhancement Transformation (BET) component to improve the representation learning of aspect, which meanwhile achieves alternative learning of aspect and context. Secondly, METNet uses a hierarchical structure to iteratively learn aspect and context to obtain better representations. Experimental results demonstrate the effectiveness of METNet for aspect-based sentiment analysis. Especially, METNet performs well in both single-aspect scenarios and multi-aspect scenarios. As future work, we will consider how to use sentiment lexicon and attention mechanism to further improve aspect representation learning.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under grant 62072169 and 61702176, and CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology, Chinese Academy of Sciences under grant CASNDST202001.

References

Erik Boiy and Marie-Francine Moens. 2009. A machine learning approach to sentiment analysis in multilingual web texts. *Information retrieval*, 12(5):526–558.

- 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. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.
- 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)*, pages 49–54.
- Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.
- 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)*, pages 946–956.
- Qiao Liu, Haibin Zhang, Yifu Zeng, Ziqi Huang, and Zufeng Wu. 2018. Content attention model for aspect based sentiment analysis. In *Proceedings of the 2018 World Wide Web Conference*, pages 1023–1032.
- Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.
- 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.
- Navonil Majumder, Soujanya Poria, Alexander Gelbukh, Md Shad Akhtar, Erik Cambria, and Asif Ekbal. 2018. Iarm: Inter-aspect relation modeling with memory networks in aspect-based sentiment analysis. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 3402–3411.
- Maria Pontiki, Dimitrios 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.
- Xiangying Ran, Yuanyuan Pan, Wei Sun, and Chongjun Wang. 2019. Learn to select via hierarchical gate mechanism for aspect-based sentiment analysis. In *IJCAI*, pages 5160–5167.
- 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.
- Jialong Tang, Ziyao Lu, Jinsong Su, Yubin Ge, Linfeng Song, Le Sun, and Jiebo Luo. 2019. Progressive self-supervised attention learning for aspect-level sentiment analysis. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 557–566.
- Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 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.
- Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2514–2523.
- Hang Yan, Bocao Deng, Xiaonan Li, and Xipeng Qiu. 2019. Tener: Adapting transformer encoder for name entity recognition. *arXiv preprint arXiv:1911.04474*.

Chao Yang, Hefeng Zhang, Bin Jiang, and Keqin Li. 2019. Aspect-based sentiment analysis with alternating coattention networks. *Information Processing & Management*, 56(3):463–478.

Matthew D Zeiler. 2012. Adadelata: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Lei Zhang, Shuai Wang, and Bing Liu. 2018. Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1253.