

# Deep Mask Memory Network with Semantic Dependency and Context Moment for Aspect Level Sentiment Classification

Peiqin Lin<sup>1</sup>, Meng Yang<sup>1,2</sup> and Jianhuang Lai<sup>1</sup>

<sup>1</sup>School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China

<sup>2</sup>Key Laboratory of Machine Intelligence and Advanced Computing(SYSU), Ministry of Education

linpq3@mail2.sysu.edu.cn, {yangm6, stsljh}@mail.sysu.edu.cn

## Abstract

Aspect level sentiment classification aims at identifying the sentiment of each aspect term in a sentence. Deep memory networks often use location information between context word and aspect to generate the memory. Although improved results are achieved, the relation information among aspects in the same sentence is ignored and the word location can't bring enough and accurate information for the analysis on the aspect sentiment. In this paper, we propose a novel framework for aspect level sentiment classification, deep mask memory network with semantic dependency and context moment (DMMN-SDCM), which integrates semantic parsing information of the aspect and the inter-aspect relation information into deep memory network. With the designed attention mechanism based on semantic dependency information, different parts of the context memory in different computational layers are selected and useful inter-aspect information in the same sentence is exploited for the desired aspect. To make full use of the inter-aspect relation information, we also jointly learn a context moment learning task, which aims to learn the sentiment distribution of the entire sentence for providing a background for the desired aspect. We examined the merit of our model on SemEval 2014 Datasets, and the experimental results show that our model achieves a state-of-the-art performance.

## 1 Introduction

Aspect level sentiment classification, which aims to identify the sentiment polarity (i.e., negative, neutral, or positive) of each aspect term in a sentence, has received much attention these years [Chen *et al.*, 2017; Liu *et al.*, 2018; Li *et al.*, 2018]. Different from the sentiment analysis for a sentence, aspect level sentiment classification conducts a finer sentiment analysis. For example, given the sentence "The price is very cheap, but the service is very poor", the sentiment polarity of the sentence is positive if the aspect is "price" while the sentiment polarity is negative if the aspect

is "service". It's obvious that the same sentence may have different sentiment polarity for different opinion aspects.

Early methods for aspect-based sentiment analysis mostly adopted supervised learning approaches with hand-crafted features [Rao and Ravichandran, 2009; Jiang *et al.*, 2011; Perez-Rosas *et al.*, 2012; Kiritchenko *et al.*, 2014; Vo and Zhang, 2015]. However, the performance of these models is highly dependent on the quality of features and feature engineering is labor intensive.

In recent years, deep neural networks have been employed to learn useful features from aspects and contexts automatically, achieving promising results on the aspect level sentiment classification task. [Dong *et al.*, 2014] learned the aspect-aware representation via semantic composition over dependency tree. [Vo and Zhang, 2015] developed a lexicon-enhanced network which used sentiment lexicon and neural pooling functions. The popular convolutional neural networks (CNNs) and long short-term memory (LSTM) have also been applied in this task. [Xue and Li, 2018] proposed a simpler and easily-parallelized model based on CNNs and gating mechanisms; [Li *et al.*, 2018] applied a transformation network (TNet) which employed CNNs to extract salient features and a component to generate target-specific representations; [Tang *et al.*, 2015] used two target-Dependent LSTM networks to separately represent text and targets, and classified the sentiment of the desired aspect by adopting a feed-forward network. However, these methods capture context information in an implicit way, and are incapable of explicitly exhibiting important context clues of an aspect.

Recently, attention mechanism and memory network have begun to be used in natural language processing tasks [Bahdanau *et al.*, 2014; Weston *et al.*, 2014; Sukhbaatar *et al.*, 2015]. Attention mechanism can capture important parts of a sentence [Wang *et al.*, 2016; Ma *et al.*, 2017; Liu *et al.*, 2018], while memory network can store context information [Tang *et al.*, 2016; Chen *et al.*, 2017]. For instance, [Liu and Zhang, 2017] applied a bidirectional LSTM network with attention and gates (BILSTM-ATT-G) to measure the importance of context; [Ma *et al.*, 2018] promoted the LSTM network with attention mechanism and common-sense knowledge; [Tay *et al.*, 2017] enabled rich dyadic interaction between aspect and document; [Fan *et al.*, 2018] proposed a convolutional memory network (Conv-Memnet) with an attention mechanism. Moreover, attention mechanism or bidirectional long short-

\*The corresponding author is Meng Yang.

term memory (BiLSTM) has also been proposed to consider the influence of other aspects in the same sentence [Majumder *et al.*, 2018; Hazarika *et al.*, 2018].

Although improved performance has been achieved by the above methods, semantic information between context and aspect and the relation between aspects in the same sentence are not well exploited. To solve the above issues, we propose a novel framework of deep mask memory network with semantic dependency and context moment (DMMN-SDCM) for aspect sentiment classification. In the proposed DMMN-SDCM, the semantic parsing information is integrated into deep memory network to guide the attention mechanism and inter-aspect learning network, which selects different parts of the context memory for different computational layers and utilizes the information provided by the nearby aspects in the sentence, respectively. To introduce the whole sentiment information, we jointly conduct a context moment learning task to explore the sentiment distribution of the entire sentence, which is able to provide a background for the desired aspect. The experiments on SemEval 2014 Datasets clearly show that our model achieves a state-of-the-art performance. The major contributions of this paper are three-fold:

1. Integrating semantic parsing information instead of location information into deep memory network to guide the attention mechanism;
2. Integrating semantic parsing information into inter-aspect modeling for utilizing the information provided by the nearby aspects in a better way;
3. Designing an auxiliary task to learn the sentiment distribution of the entire sentence, which can provide a background for the sentiment analysis of the desired aspect;

## 2 Related Work

The deep learning models with attention mechanism and memory network has been applied to the aspect-level sentiment classification, with promising performance reported.

Attention mechanism has been widely applied in sentiment classification. [Wang *et al.*, 2016] proposed the attention-based LSTM with aspect embedding (ATAE-LSTM), which firstly applied attention mechanism to aspect level sentiment classification by simply concatenating the aspect vector into the sentence hidden representations and achieved a good performance. [Ma *et al.*, 2017] developed an interactive attention network (IAN) to learn the representations of aspects and contexts, respectively. [Liu *et al.*, 2018] applied sentence-level content attention mechanism and learned the weights of the memory with two gated recurrent unit (GRU) networks.

Memory network is also used to capture the context information. [Tang *et al.*, 2016] first used deep memory network (MemNet), in which attention mechanism with multiple computational layers is applied over an external memory and location attention is integrated into content location. [Chen *et al.*, 2017] proposed a recurrent attention network which used a BiLSTM model to generate the memory and combined all attention results in a nonlinear way by using a GRU model.

To utilize the information of inter-aspect relation, some effective inter-aspect relation modeling methods have been proposed. [Majumder *et al.*, 2018] applied attention mechanism

to model the relation between aspects while [Hazarika *et al.*, 2018] applied a BiLSTM network.

Although deep memory network and inter-aspect modeling have achieved promising results on aspect level sentiment classification, there are still some problems unsolved.

Firstly, the semantic information of aspect in the sentence can't be well exploited. The deep memory network approaches [Tang *et al.*, 2016; Chen *et al.*, 2017], which generate location information according to the absolute distance from each context word to aspect, can't capture important related words in semantic analysis. For example, given the sentence "Food is good but service is bad" and the aspect word "service", the context word "good" is considered to be as important as the context word "bad" by the traditional methods.

Secondly, the global information of the sentence, including the sentiment polarity of the entire sentence and the relation between aspects, isn't well exploited in previous methods. For example, given the sentence "service and food are good.", it's easy to know the sentiment results of "service" and "food" if we know the sentiment of the sentence is positive and the relation between the two aspects is coordination.

## 3 Model

To address the issues above, we proposed a novel model for aspect level sentiment classification, deep mask memory network with semantic dependency and context moment (DMMN-SDCM). Based on deep memory network, we introduce the information of semantic parsing to guide the attention mechanism and effectively learn the information provided by other aspects in the same sentence. Meanwhile the context moment embedded in the sentiment of entire sentence is designed to provide a background for the desired aspect. In the following, we will first give an overview of the proposed DMMN-SDCM, and then detail the semantic-dependency-mask attention, the inter-aspect semantic modeling and the context-moment sentiment learning. In our paper, the operation of  $(x, y)$  and  $(x, y, z)$  are used to denote the concatenation of two vectors (e.g.,  $x$  and  $y$ ) and three vectors (e.g.,  $x$ ,  $y$ , and  $z$ ), respectively.

### 3.1 Overview of DMMN-SDCM

The architecture of DMMN-SDCM is shown in Fig. 1, which is composed of six modules: embedding module, memory building module, semantic-dependency-mask attention module, inter-aspect semantic modeling module, context-moment sentiment learning module, and classification module.

In our model, as the task setting of aspect level sentiment classification [Li *et al.*, 2018], the aspects appearing in the sentence have been extracted and identifying the sentiment of the given aspects is the focus. To avoid aspect information from interfering with context modeling, we replace each aspect, which may be word or a phrase, with a placeholder.

The embedding module uses a word embedding lookup table, which can be pre-trained by unsupervised methods like Word2Vec [Mikolov *et al.*, 2013], Glove [Pennington *et al.*, 2014] or FastText [Joulin *et al.*, 2016; Bojanowski *et al.*, 2016], to convert most words to theirs word vectors, then initialize other out-of-vocabulary words by sampling from the uniform distribution.

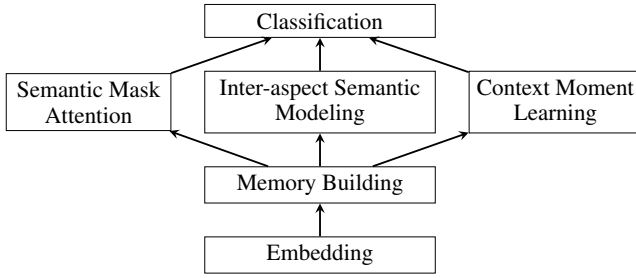


Figure 1: The overall architecture of DMMN-SDCM.

To build an external memory, the memory building module applies a BiLSTM network, which can capture the relationship between words in a sentence. After modeling, the final memory generated is  $\{\mathbf{m}_1^*, \dots, \mathbf{m}_i^*, \dots, \mathbf{m}_n^*\}$ , where  $n$  stands for the length of sentence and  $\mathbf{m}_i^* = (\overrightarrow{\mathbf{h}}_i, \overleftarrow{\mathbf{h}}_i)$ , with  $\overrightarrow{\mathbf{h}}_i$  and  $\overleftarrow{\mathbf{h}}_i$  denoting the hidden states generated by the forward and the backward LSTMs of BiLSTM, respectively.

The semantic-dependency-mask attention module extracts semantic parsing information, and then selects different slices of memory dynamically for different computational layers to guide the attention mechanism. The output of the module for the desired aspect is denoted as  $\mathbf{v}_{sd}$ .

To utilize the information of other aspects in the same sentence, the inter-aspect semantic modeling module and the context-moment sentiment learning module are designed. The former module uses semantic dependency information and attention mechanism to get the useful information of other aspects, while the latter module applies a context moment learning task to learn the relation between all aspects in the entire sentence. The outputs of the two modules are denoted as  $\mathbf{v}_{im}$  and  $\mathbf{v}_{cm}$ .

With the last three modules, we can get the sentiment representation of the desired aspect:

$$\mathbf{v} = \{\mathbf{v}_{sd}, \mathbf{v}_{im}, \mathbf{v}_{cm}\} \quad (1)$$

The classification module adopt a feed-forward network to project  $\mathbf{v}$  into the space of the targeted  $C$  classes.:

$$\mathbf{y} = \text{softmax}(\mathbf{W}_c \mathbf{v} + \mathbf{b}_c) \quad (2)$$

where  $\mathbf{y} = [y_1, \dots, y_i, \dots, y_C] \in R^C$  is the estimated probability, and  $\mathbf{W}_c$  and  $\mathbf{b}_c$  are weight matrix and bias, respectively. The model is finally trained by minimizing the sum of the cross entropy, the  $L_2$  regularization term and the context moment learning loss:

$$L = \sum_{s \in S} \left( - \sum_{t \in T} \sum_{i=1}^C g_i^t \log y_i^t + \lambda_m l_m(s) \right) + \lambda \|\theta\|_2^2 \quad (3)$$

where  $S$  is the sentence set of the training data and  $T$  is the aspect set of the sentence  $s$ .  $\mathbf{g}^t = [g_1^t, \dots, g_i^t, \dots, g_C^t] \in R^C$  is the ground truth for aspect  $t$ , which is represented by a one-hot vector where the element for the correct polarity is 1.  $\mathbf{y}^t = [y_1^t, \dots, y_i^t, \dots, y_C^t] \in R^C$  is the estimated probability for aspect  $t$ .  $\lambda$  is the weight of  $L_2$  regularization term, and  $\lambda_m$  is the weight of the context moment learning loss. The term of  $l_m(s)$  is the context moment learning loss for the sentence  $s$ , which will be detailed in the following section.

### 3.2 Semantic-Dependency-Mask Attention

In this section, we present the semantic-dependency-mask attention, as shown in the right part of Fig. 2. To guide the attention mechanism, the module masks the context memory for different computational hops according to the semantic dependency parsing tree. At each computational hop, the module applies attention mechanism between the representation of aspect and every memory unit. The results of multiple attention layers are combined in a nonlinear way.

We use a open-source library spaCy<sup>1</sup>, whose parsing model is a blend of recent methods, to generate the dependency parsing tree. In a dependency tree, the distances from context word to different aspects are different, so that we can get different semantic parsing information for different aspects.

Depending on the distance between each context words and the aspect, we can extract semantic parsing information from the dependency tree of the sentence. For different layers in deep memory network, the model selects different parts of memory depending on semantic parsing information. At  $l$ -th layer, each memory unit masked with semantic dependency parsing information can be represented as follows:

$$\mathbf{m}_i^l = \begin{cases} \mathbf{m}_i^*, & \text{if } \text{dist}(w_i, w_t) \leq l \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where  $\mathbf{m}_i^*$  stands for the original memory unit generated by the BiLSTM, and  $\text{dist}(w_i, w_t)$  is the length of the path from the aspect word  $w_t$  to the context word  $w_i$  in the semantic dependency tree of the sentence. Therefore, the final masked memory generated at  $l$ -th layer is  $\{\mathbf{m}_1^l, \dots, \mathbf{m}_i^l, \dots, \mathbf{m}_n^l\}$ .

With masked memory, we can get the normalized attention score  $\alpha_i^l$  of each memory unit at  $l$ -th layer  $\mathbf{m}_i^l$  as follows:

$$\alpha_i^l = \text{softmax}(\mathbf{W}_{AL}^l (\mathbf{m}_i^l, \mathbf{v}_{sd}^{l-1}, \mathbf{v}_t) + \mathbf{b}_{AL}^l) \quad (5)$$

where  $\mathbf{W}_{AL}^l$  and  $\mathbf{b}_{AL}^l$  denote weight matrices and biases of the  $l$ -th layer, respectively, and  $\mathbf{m}_i^l$  and  $\mathbf{v}_{sd}^{l-1}$  stand for the memory unit and the output from the previous layer, respectively. Here  $\mathbf{v}_t$  denotes the aspect representation, which is generated by adopting a BiLSTM layer and an attention layer.

And the attention output  $i_t^{AL}$  is calculated as:

$$i_t^{AL} = \sum_{i=1}^n \alpha_i^l \mathbf{m}_i^l \quad (6)$$

To generate the final representation of the  $l$ -th layer, we adopt a linear layer as well as a transform layer, which introduces a gating function to control the passed proportions of the transformed features and the input features [Srivastava et al., 2015]:

$$\mathbf{v}_{sd}^l = (\mathbf{W}_o i_t^{AL} + \mathbf{b}_o) * \mathbf{T} + \mathbf{v}_{sd}^{l-1} * (1 - \mathbf{T}) \quad (7)$$

where  $\mathbf{W}_o$  and  $\mathbf{b}_o$  are weight matrices and biases of the linear layer, respectively. And the gate  $\mathbf{T}$  as follows:

$$\mathbf{T} = \sigma(\mathbf{W}_t \mathbf{v}_{sd}^{l-1} + \mathbf{b}_t) \quad (8)$$

where  $\sigma$  stands for the *sigmoid* activation function, and  $\mathbf{W}_t$  and  $\mathbf{b}_t$  are weight matrices and biases, respectively.

Finally, the output of the last layer will be taken as the output of the semantic-dependency-mask attention module  $\mathbf{v}_{sd}$ .

<sup>1</sup><https://spacy.io/>

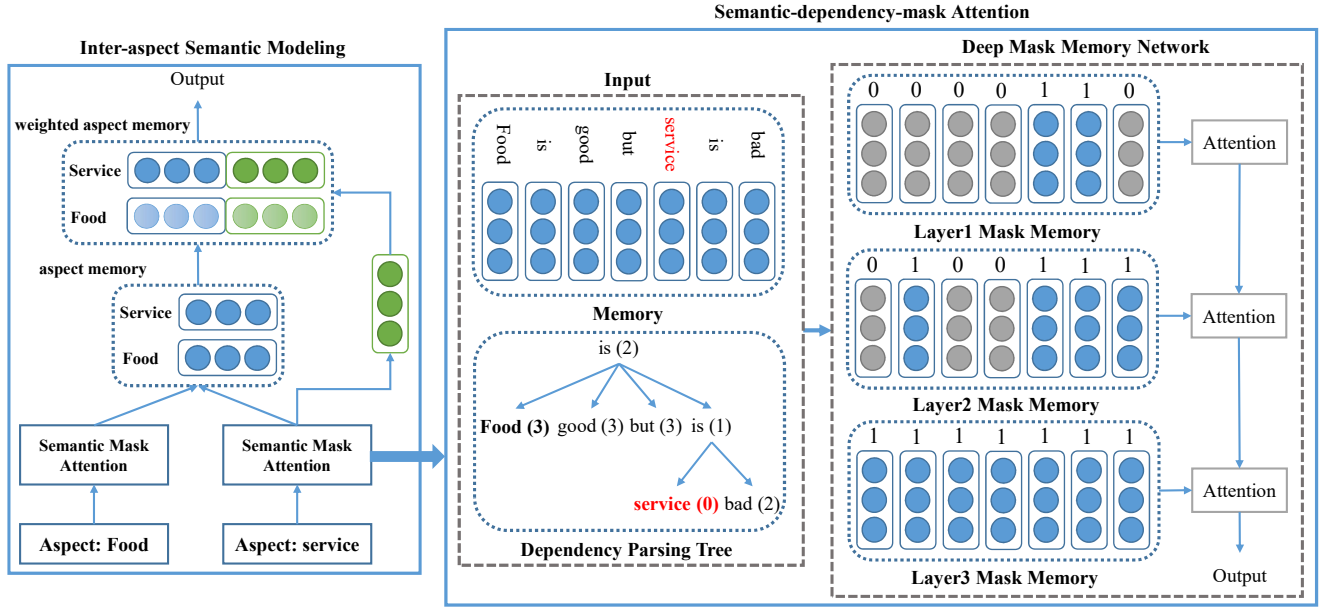


Figure 2: Architecture of inter-aspect semantic modeling and semantic-dependency-mask attention. The left part of the semantic mask attention module illustrates the input of the deep mask memory network, namely the memory  $\{m_1^*, \dots, m_i^*, \dots, m_n^*\}$  and the dependency parsing tree whose nodes are in the form of  $word(dist(w_i, w_t))$ , where  $dist(w_i, w_t)$  is the distance from the aspect  $w_t$  to the context word  $w_i$  in the dependency tree. The right part shows how the deep mask memory network works. Attention mechanism is only applied on the selected memory slices remaining colored in the figure.

### 3.3 Inter-Aspect Semantic Modeling

To model the inter-aspect relation, we propose a novel framework which integrates semantic dependency information into attention mechanism (shown in the left part of Fig. 2). After applying the semantic mask attention, we can get  $p$  aspect-aware representations for all  $p$  aspects in the sentence. Then we stack the representations as the aspect memory according to the order of occurrence of their corresponding aspect-terms. The aspect memory can be represented as follows:

$$\mathbf{M} = \{\mathbf{v}_{sd_1}, \mathbf{v}_{sd_2}, \dots, \mathbf{v}_{sd_p}\} \quad (9)$$

For different aspects in the same sentence, their original aspect memories are same. To introduce the information of the desired aspect to the aspect memory, we concatenate every memory unit and the representation of the desired aspect  $\mathbf{v}_{sd_t}$ , and weight the memory according to the semantic dependency information.

$$\mathbf{M}' = \{\mathbf{v}'_{sd_1}, \mathbf{v}'_{sd_2}, \dots, \mathbf{v}'_{sd_p}\} \quad (10)$$

where  $\mathbf{v}'_{sd_i} = \lambda_i * (\mathbf{v}_{sd_i}, \mathbf{v}_{sd_t})$  and  $\lambda_i$  is the semantic dependency weight, which is calculated as follows:

$$\lambda_i = 1 - \frac{dist(w_i, w_t)}{depth} \quad (11)$$

where  $dist(w_i, w_t)$  is the semantic distance from the current aspect  $w_i$  to the desired aspect  $w_t$  and  $depth$  stands for the depth of the dependency tree.

Then we apply the attention mechanism and get the final representation  $\mathbf{v}_{im}$ .

$$\beta_i = softmax(\mathbf{W}_{im} \mathbf{v}'_{sd_i} + \mathbf{b}_{im}) \quad (12)$$

$$\mathbf{v}_{im} = \sum_i^p \beta_i * \mathbf{v}'_{sd_i} \quad (13)$$

where  $\mathbf{W}_{im}$  and  $\mathbf{b}_{im}$  are weight matrices and biases.

### 3.4 Context-Moment Sentiment Learning

For a sentence with multiple aspects, there are generally some sentiment relations between aspects, like comparisons and coordinations. It will be helpful to identify the aspect sentiment polarity if the relation between aspects is introduced to the model, since these kind of relations can provide the sentence sentiment background for sentiment classification task. In this section, we propose a context moment learning task to capture the distribution of the sentence sentiment, and design a context-moment sentiment learning module, as shown in Fig. 3.

Let  $-1$ ,  $0$  and  $1$  denote the sentiment polarities negative, neutral and positive, then we use moment to portray the sentiment distribution of aspects in sentence  $s$ , and the moment is defined as follows:

$$\mu_i = \mathbf{E}((\mathbf{x} - \mu)^i) \quad (14)$$

where  $\mathbf{E}$  is the expectation operator,  $\mathbf{x}$  is the random variable of the sentiment distribution,  $\mu$  is the mean of the distribution, and  $i$  stands for the rank of the moment. It's easy to know  $\mu_i \in [-1, 1]$  if the rank is odd while  $\mu_i \in [0, 1]$  if even. Therefore we normalize all the moments to  $[0, 1]$ .

To illustrate the context moment descriptors clearer, we give some examples shown in Table. 1. From the table, we

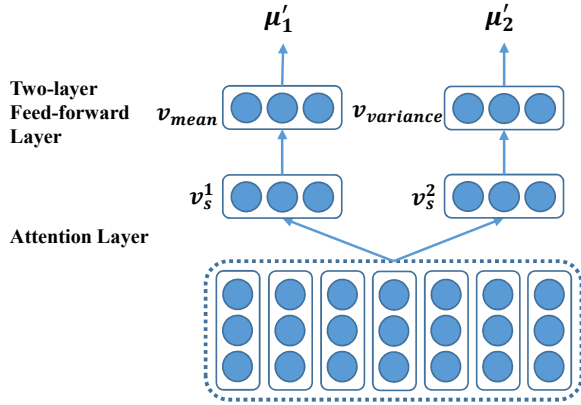


Figure 3: Architecture of context-moment sentiment learning.

Sentences	Distribution	Mean	Var
<b>Food</b> and <b>service</b> are good.	[1, 1]	1	0
<b>Food</b> and <b>service</b> are bad.	[-1, -1]	0	0
<b>Food</b> are good, <b>service</b> are bad.	[1, -1]	0.5	1

Table 1: Context moment calculation examples with two aspects. The normalized value of mean and variance, namely the first and second moment are given in the table.

observe that the greater the mean, the more positive the overall sentiment of the sentence. In addition, the closer the variance is to 1, the closer the relation between aspects is to comparison, while the closer the variance is to 0, the closer the relation is to coordination. Therefore, the context moment can describe the overall sentiment of the sentence and the relation between aspects.

To learn the estimation of context moments like mean and variance, the module applies an attention layer and a two-layer feed-forward network. And then a context-moment loss for a sentence  $s$  is defined to help the optimization of aspect sentiment classification task:

$$l_m(s) = \sum_{i=1}^k \mu_i \ln(\mu'_i) + (1 - \mu_i) \ln(1 - \mu'_i) \quad (15)$$

where  $k$  stands for the greatest rank of the moment used,  $\mu_i$  is the value of the  $i$ th moment while  $\mu'_i$  is the estimate value.

To portray the sentiment distribution of the overall sentence, we use the first moment  $\mu_1$  and the second moment  $\mu_2$  as the learning goal of context-moment learning. To learn these moments, we firstly apply attention mechanism to get the sentence representation  $v_s^1$  and  $v_s^2$  for the first and second moment, respectively, then use a two-layer fully connected network can get the estimates of moments. For example, we can get the mean representation of the sentence:

$$v_{mean} = W_{\mu_1} v_s^1 \quad (16)$$

Then the estimation of the first moment is obtained:

$$\mu'_1 = \text{sigmoid}(W_{\mu_1}(v_{mean}; v_{global})) \quad (17)$$

where  $v_{global}$  is calculated by taking the average of the all  $p$  representations of the aspects generated by the semantic mask

Dataset	Positive		Negative		Neutral	
	train	test	train	test	train	test
Laptop	994	341	870	128	464	169
Restaurant	2164	728	807	196	637	196

Table 2: Details of the experimental datasets.

attention module:

$$v_{global} = \frac{1}{p} \sum_i^p v_{sd_i} \quad (18)$$

By the same way, we can get the variance representation  $v_{variance}$  and the estimate of the second moment  $\mu'_2$ .

Finally, the module will generate the corresponding representations of the sentence for each moment, which are embedded to the classification module for the analysis of the aspect sentiment. Specifically, The mean representation and the variance representation will be concatenated as the final output of the module:

$$v_{cm} = (v_{mean}, v_{variance}) \quad (19)$$

## 4 Experiment

### 4.1 Experimental Setup

To verify the effectiveness of DMMN-SDCM, we do experiments on SemEval 2014 Task 4 [Pontiki *et al.*, 2014]. The SemEval 2014 reviews dataset consists of Laptop dataset and Restaurant dataset. The reviews are labeled with four sentiment polarities: positive, neutral, negative and conflict. We remove conflict category as the number of conflict samples is very small and make the dataset extremely unbalanced. The details of the experimental datasets are shown in Table. 2

The embedding table used in experiments is pre-trained by Glove [Pennington *et al.*, 2014]. The out-of-vocabulary words are initialized by sampling from the uniform distribution  $U(-0.25, 0.25)$ . The dimension of word vectors is 300.

We employ Adam Optimizer [Kingma and Ba, 2014] to train our model. The initial learning rate is 0.01, the weight of  $L_2$  regularization term is 0.0001, the weight of context moment loss is 1.5 and the dropout rate is 0.5. The dimension of LSTM hidden states and output representation is 50. The evaluation metrics are Accuracy and Macro-F1.

### 4.2 Comparison with Baselines

To evaluate the performance of DMMN-SDCM, we compare our model to some baseline methods: TD-LSTM [Tang *et al.*, 2015], ATAE-LSTM [Wang *et al.*, 2016], MemNet [Tang *et al.*, 2016], IAN [Ma *et al.*, 2017], BILSTM-ATT-G [Liu and Zhang, 2017], RAM [Chen *et al.*, 2017], Conv-Memnet [Fan *et al.*, 2018] and TNet [Li *et al.*, 2018]. The results of performance comparison is shown in Table. 3.

From the experimental results, we can observe that DMMN-SDCM outperforms all baselines in both Restaurant reviews and Laptop reviews. Our model has the best performance through integrating semantic dependency information and context moment information into deep memory network. Semantic dependency information can not only provide dynamic and reasonable memory for each attention layer, but

Models	Laptop		Restaurant	
	ACC	Macro-F1	ACC	Macro-F1
TD-LSTM	71.83	68.43	78.00	66.73
ATAE-LSTM	68.65	62.45	77.23	64.95
MemNet	70.33	64.09	78.16	65.83
IAN	72.10	67.48	77.95	67.90
BiLSTM-ATT-G	74.37	69.90	80.38	70.78
RAM	75.01	70.51	79.79	68.86
Conv-Memnet	76.37	72.10	78.26	68.38
TNet	76.54	71.75	80.79	71.27
DMMN-SDCM	<b>77.59</b>	<b>73.61</b>	<b>81.16</b>	<b>71.50</b>

Table 3: Comparison with baselines.

Contained Modules	ACC	Macro-F1
$M_{sd}$	75.08	71.68
$M_{sd} + M_{im}$	76.80	72.82
$M_{sd} + M_{cm}$	76.18	71.91
$M_{sd} + M_{im} + M_{cm}$	<b>77.59</b>	<b>73.61</b>

Table 4: Effects of each module on the Laptop dataset.  $M_{sd}$ ,  $M_{im}$  and  $M_{cm}$  represents the mask attention module, the inter-aspect modeling module and the context-moment module, respectively.

also guide the inter-aspect attention mechanism to capture the important information of other aspects. Moreover, the context-moment learning captures the sentence sentiment distribution, which is helpful for aspect sentiment classification.

### 4.3 Effects of Each Module

To examine the effectiveness of each module, we conduct extensive experiments and the results are shown in Table. 4.

From Table. 4, integrating semantic dependency information into deep memory is effective since the model which only contains the mask attention module outperforms RAM [Chen *et al.*, 2017], which applies position-weighting attention. Moreover, the two modules, context moment learning and inter-aspect modeling, improve the performance of the model too, since both of them utilize the relation between aspects in the sentence. With the three creative modules, our model outperforms all the baselines.

### 4.4 Effects of Inter-Aspect Modeling

To verify if semantic-dependency weighting method works, we conduct experiments and the results are shown in Table. 5.

We can observe that weighting strategy can get improved performance whatever method the model uses to model inter-aspect relation. In addition, we can see that attention mechanism [Majumder *et al.*, 2018] is more effective than BiLSTM

Models	ACC	Macro-F1
Without Inter-aspect Modeling	76.18	71.91
BiLSTM	75.08	70.40
BiLSTM + Semantic Weighting	75.24	70.83
Attention	75.55	71.73
Attention + Semantic Weighting	<b>77.59</b>	<b>73.61</b>

Table 5: Effects of inter-aspect modeling on the Laptop dataset.

Models	ACC	Macro-F1
Without Moment Learning	76.80	72.82
Using Mean	75.55	70.86
Using Variance	74.45	71.67
Using Mean and Variance	<b>77.59</b>	<b>73.61</b>
Using Mean, Variance & Skewness	75.86	71.92

Table 6: Effects of moment ranks on the Laptop dataset. “Mean”, “Variance” and “Skewness” are the first, second and third moments.

[Hazarika *et al.*, 2018] for modeling inter-aspect relation.

### 4.5 Effects of Moment Ranks

Introducing different context moments to the model may have different effects. We evaluate our model with using different context moments, and the results are shown in Table. 6.

From the table, we can see that either using the first moment only or using the second moment only can’t achieve good performance, for the reason that insufficient information is obtained. In addition, introducing the higher-rank moment isn’t effective too, as it may bring interference information as the aspect sentiment distribution is not complex enough.

As the experimental results show, using the first and the second moment can perform best. The mean representation can model the sentiment of the whole sentence while the variance representation can model the inter-aspect relation.

## 5 Conclusion

In this paper, we design a deep mask memory network with semantic dependency and context moment (DMMN-SDCM) which integrates semantic parsing information and context moment learning into deep memory network for the first time. With the semantic dependency, we proposed a more discriminative attention scheme, which effectively selects different parts of the context memory for different computational layers, and presented an effective method to model the inter-aspect relation. The sentiment distribution of the entire sentence is also encoded by using a context moment for the first time, which guides the deep memory network to learn an effective feature. We have conducted extensive experiments on SemEval 2014 review datasets, and the experiment results clearly show that our model performance is state-of-the-art.

## Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (Grant no.61772568), the Key Areas Research and Development Program of Guangdong (Grant no.2018B010109007), the Guangzhou Science and Technology Program (Grant no. 201804010288), and the Fundamental Research Funds for the Central Universities (Grant no.18lgzd15).

## References

- [Bahdanau *et al.*, 2014] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.
- [Bojanowski *et al.*, 2016] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2016.
- [Chen *et al.*, 2017] Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. Recurrent attention network on memory for aspect sentiment analysis. In *EMNLP*, pages 452–461, 2017.
- [Dong *et al.*, 2014] Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. Adaptive recursive neural network for target-dependent twitter sentiment classification. In *ACL*, volume 2, pages 49–54, 2014.
- [Fan *et al.*, 2018] Chuang Fan, Qinghong Gao, Jiachen Du, Lin Gui, Ruifeng Xu, and Kam-Fai Wong. Convolution-based memory network for aspect-based sentiment analysis. In *SIGIR*, pages 1161–1164. ACM, 2018.
- [Hazarika *et al.*, 2018] Devamanyu Hazarika, Soujanya Poria, Prateek Vij, Gangeshwar Krishnamurthy, Erik Cambria, and Roger Zimmermann. Modeling inter-aspect dependencies for aspect-based sentiment analysis. In *NAACL*, volume 2, pages 266–270, 2018.
- [Jiang *et al.*, 2011] Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. Target-dependent twitter sentiment classification. In *ACL*, pages 151–160. ACL, 2011.
- [Joulin *et al.*, 2016] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016.
- [Kingma and Ba, 2014] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [Kiritchenko *et al.*, 2014] Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *SemEval*, pages 437–442, 2014.
- [Li *et al.*, 2018] Xin Li, Lidong Bing, Wai Lam, and Bei Shi. Transformation networks for target-oriented sentiment classification. *arXiv preprint arXiv:1805.01086*, 2018.
- [Liu and Zhang, 2017] Jiangming Liu and Yue Zhang. Attention modeling for targeted sentiment. In *EACL*, volume 2, pages 572–577, 2017.
- [Liu *et al.*, 2018] Qiao Liu, Haibin Zhang, Yifu Zeng, Ziqi Huang, and Zufeng Wu. Content attention model for aspect based sentiment analysis. In *WWW*, pages 1023–1032. WWW, 2018.
- [Ma *et al.*, 2017] Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. Interactive attention networks for aspect-level sentiment classification. *arXiv preprint arXiv:1709.00893*, 2017.
- [Ma *et al.*, 2018] Yukun Ma, Haiyun Peng, and Erik Cambria. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm. In *AAAI*, 2018.
- [Majumder *et al.*, 2018] Navonil Majumder, Soujanya Poria, Alexander Gelbukh, Md Shad Akhtar, Erik Cambria, and Asif Ekbal. Iarm: Inter-aspect relation modeling with memory networks in aspect-based sentiment analysis. In *EMNLP*, pages 3402–3411, 2018.
- [Mikolov *et al.*, 2013] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014.
- [Perez-Rosas *et al.*, 2012] Veronica Perez-Rosas, Carmen Banea, and Rada Mihalcea. Learning sentiment lexicons in spanish. In *LREC*, volume 12, page 73, 2012.
- [Pontiki *et al.*, 2014] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. Semeval-2014 task 4: Aspect based sentiment analysis. *SemEval*, pages 27–35, 2014.
- [Rao and Ravichandran, 2009] Delip Rao and Deepak Ravichandran. Semi-supervised polarity lexicon induction. In *EACL*, pages 675–682. ACL, 2009.
- [Srivastava *et al.*, 2015] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Highway networks. *arXiv preprint arXiv:1505.00387*, 2015.
- [Sukhbaatar *et al.*, 2015] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *NIPS*, pages 2440–2448, 2015.
- [Tang *et al.*, 2015] Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. Effective lstms for target-dependent sentiment classification. *arXiv preprint arXiv:1512.01100*, 2015.
- [Tang *et al.*, 2016] Duyu Tang, Bing Qin, and Ting Liu. Aspect level sentiment classification with deep memory network. *arXiv preprint arXiv:1605.08900*, 2016.
- [Tay *et al.*, 2017] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. Dyadic memory networks for aspect-based sentiment analysis. In *CIKM*, pages 107–116. ACM, 2017.
- [Vo and Zhang, 2015] Duy-Tin Vo and Yue Zhang. Target-dependent twitter sentiment classification with rich automatic features. In *IJCAI*, pages 1347–1353, 2015.
- [Wang *et al.*, 2016] Yequan Wang, Minlie Huang, Li Zhao, et al. Attention-based lstm for aspect-level sentiment classification. In *EMNLP*, pages 606–615, 2016.
- [Weston *et al.*, 2014] Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. *Eprint Arxiv*, 2014.
- [Xue and Li, 2018] Wei Xue and Tao Li. Aspect based sentiment analysis with gated convolutional networks. *arXiv preprint arXiv:1805.07043*, 2018.