# MAN: Mutual Attention Neural Networks Model for Aspect-Level Sentiment Classification in SIoT

Nan Jiang, *Member, IEEE,* Fang Tian, Jin Li, *Member, IEEE,* Xu Yuan, *Member, IEEE,* and Jiaqi Zheng, *Member, IEEE*

***Abstract*—Text sentiment classification is critical of importance to improve the autonomous decision making and communication ability among object peers in Social Internet of Things (SIoT). To classify sentiment polarity on a fine-grained level, aspect-level sentiment classification has become a promising direction in recent years. However, the existing solutions typically ignored the mutual information between sentences and their respective aspect terms while generally performing sentiment classification by using simple attention mechanism. Thus, the relevant results seem to be unpromising. We aim to develop a novel neural-network-based model, by relying on the NLP model for rich feature extraction, called mutual attention neural networks (abbreviated as MAN) to conduct the aspect-level sentiment classification tasks in this paper. Compared with previous work, our proposed MAN model takes advantage of the bidirectional long short-term memory (Bi-LSTM) networks to obtain semantic dependence of sentences and their respective aspect terms, while learning the sentiment polarities of aspect terms in sentences by proposing mutual attention mechanism. To evaluate the performance of MAN, we conduct our experiments on three real-world datasets, i.e., LAPTOP, REST, and TWITTER. The experimental results show that our proposed MAN model has significantly performance improvements when compared to several existing models, in terms of aspect-level sentiment classification.**

***Index Terms*—Natural Language Processing, Social Internet of Things, Sentiment Classification, Mutual Attention Mechanism**

## I. Introduction

THE Internet of Things (IoT), referring to the "Internet connected by all things", is an extension of the Internet by combining various information sensing devices, computing devices, objects, and people with Internet to form a large-sized operational networks [1]. It can realize the interconnection of human, machines, and things at any time and location. On the other hand, social networks [2] (such as Facebook, Twitter, and Weibo, etc.), as the important Internet applications, have become prevalent and are playing critical roles in our daily life, by providing communication and interactive services for online aggregated users in various forms with certain social relationships or common interests. Recently, the research community initialed an emerging paradigm with the integration of social networks and IoT, named the Social Internet of Things (SIoT) [3], with the aim to leverage the social users' behaviors or opinions to enhance the IoT devices' functionality. Fig. 1 illustrates the framework of SIoT. SIoT relies on the topology of social networks and their entities, represented by intelligent hardware and users, utilizes the social relationships to construct the effective models that can capture social networks characteristics. Such characteristics offer the valuable information for human's activities and behaviors, which thus can be well utilized by SIoT networks, with perceptual monitoring technology, to make the intelligent decision for their network deployment or service enhancements. Thus, SIoT can take advantage of social networks topology and information to enhance the user-friendliness and connectivity of IoT networks. In addition, it can improve intelligence and context awareness so as to support autonomous decision making and communication among object peers.

In SIoT, social media platforms typically generate large amounts of data, which are the valuable resources to help people or machines make decision for the control or monitoring components by analyzing the inherent opinions or sentiment information. Such efforts are critical of importance as they have the potential of bringing significant benefits to society and national lives. One critical task in SIoT is to capture human-to-human friendliness and perception by analyzing user-related services. This requires us to develop effective solutions to extract features from the natural language of user-related services and analyze such features to mine users' inherent meanings. Natural language processing (NLP) [4] and computational linguistics [5], [6] are powerful techniques that can be used by us, which have been applied to many areas successfully, such as text classification, sentiment analysis, question and answer system, machine translation, and named entity recognition, etc. Through these applications, SIoT can automatically extract users' inherent meanings, which can offer valuable information to human or smart devices to make decisions for user-related services.

To understand user-related services, sentiment analysis from the social networks is a necessary and fundamental task with the aim to mine users' opinion in SIoT. We aim to analyze user's sentiment information to understand users' opinions so

This work is supported by National Natural Science Foundation of China under Grant No. 61462028 and 61962022, Jiangxi Provincial Cultivation Program for Academic and Technical Leaders of Major Subjects under Grant No.20172BCB22017,Excellent Scientific and Technological Innovation Teams of Jiangxi Province under Grant No. 20181BCB24009. *Nan Jiang and Jin Li are corresponding authors*.

Nan Jiang and Fang Tian are with Department of Internet of Things, East China Jiaotong University, Nanchang 330013, P.R.China(email: jiangnan@ecjtu.edu.cn).

Jin Li is with School of Computer Science, Guangzhou University, Guangzhou 510631, P.R.China.

Xu Yuan is with School of Computing and Informatics, University of Louisiana at Lafayette, LA, USA 70503.

Jiaqi Zheng is with State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, P.R.China.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2963927, IEEE Internet of Things Journal

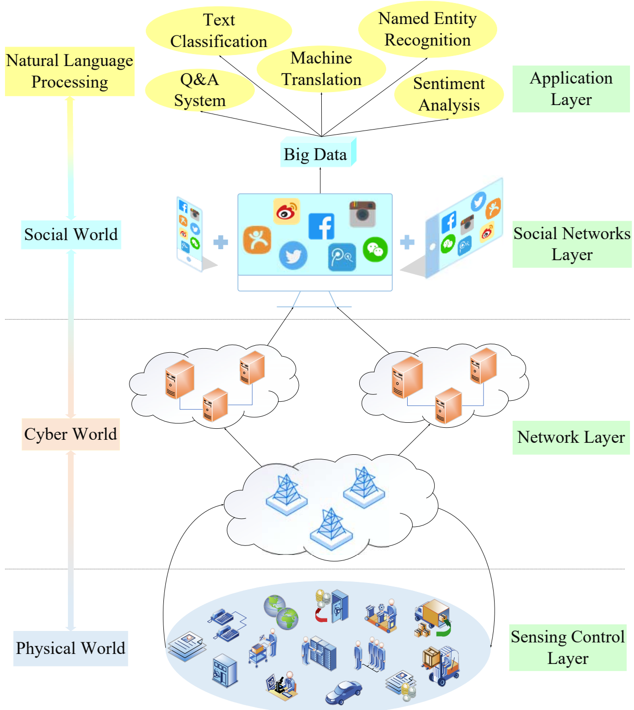



Fig. 1. The framework of SIoT.

that they can be well utilized by SIoT in this paper. According to the coarse or fine division of the grain, there are three different levels of sentiment analysis, i.e., document level, sentence level, and aspect level. Document-level sentiment classification specifies an overall sentiment polarity to determine whether the article is conveying an overall positive, negative or neutral opinions. Sentence-level sentiment classification has a finer grain than document-level, it determines the sentiment polarity of the entire sentence. Aspect-level sentiment classification has the finest grain of them, it targets to classify the sentiment polarity of each disparate aspect in one sentence.

If there are multiple aspects existing in one sentence, the aspect-level sentiment classification model can capture the sentiment polarities of disparate aspects. For instance, in the sentence "The portions are small but being that the food was so good makes up for that.", there are two aspects: "portions" and "food". Because the sentence expresses a negative attitude toward "portions", its correct sentiment polarity is negative. Additionally, this sentence expresses a positive attitude toward "food", so its anticipated sentiment polarity is positive. Such a polarity identification work cannot be tackled by the traditional sentiment model, i.e., document-level and sentence-level, because when multiple aspects appear in one sentence, the sentiment of entire sentence may consist of negative, positive, and neutral polarities. This is extremely important to judge the polarity of a sentence by using aspect-level sentiment classification model rather than the other two models, to capture the correct polarity in the appearance of positive negative, positive, or neutral polarities. The general sentiment classification tasks can make polarity mistakes without taking into account the multiple aspects. Overall, the aspect-level sentiment classification has the superior advantages over both document-level and sentence-level sentiment classification.

Many efforts have been put on sentiment classifications. For example, [7] has conducted a systematic and comprehensive study of sentiment classification on film reviews by using the supervised machine learning algorithms. In [8], the sentiment classification method for online comment texts has been proposed by using unsupervised learning algorithm based on the sentiment lexicon and manual decision rules. In [9], an unsupervised deep neural network is proposed and works effectively, which attracts both industry and research community's attentions to explore the deep learning approaches on the sentiment classification. On the other hand, some work has focused on the NLP while integrating the popular deep neural networks, i.e., long short-term memory (LSTM) networks models [10] and convolutional neural networks (CNN) models [11]. When conducting the sentiment classification tasks, the sentences are represented with features vectors to provide the affluent information for the training of deep neural networks. Even the existing work has explored the sentiment information to some extent, they mainly concentrate on the document level, sentence level, or simple attention mechanisms. The aspect-level sentiment classification with the consideration of the mutual attention between sentences and their respective aspect terms, which have rich information to reflect a sentence's sentiment information, is not explored yet.

We propose a new framework in this paper, named mutual attention neural networks (abbreviated as MAN), to conduct the aspect-level sentiment classification tasks. This framework is based on LSTM and attention mechanisms, which aims to acquire the important information of the sentences and their respective aspect terms while making full use of the mutual information between them. Thus, such a framework can significantly improve sentiment classification results. Mutual attention is gained through the following steps. We first obtain the initial representations of the sentence and the aspect term by calculating the average of the two hidden matrices generated by Bi-LSTMs, and then multiply these types of matrices. Two softmax functions are used to calculate the probability distribution in each column and row of the matrix that was generated in the previous step. In the end, we calculate the average again to acquire the mutual information between the sentence and the aspect term, and then perform the multiplication operations. To evaluate the performance of our proposed solution, we do experiments on three real-world datasets: LAPTOP, REST, and TWITTER. The experimental results demonstrate that our proposed solution can improve the performance of sentiment classification significantly when compared to multiple existing solutions.

We summarize the contributions of this paper as follows:

- A new model named mutual attention neural networks (abbreviated as MAN) is proposed in this paper, aiming to extract the mutual attention information between sentences and their respective aspect terms to conduct aspect-level sentiment classification tasks. Such a MAN model can correctly classify the sentiment polarity when there are various aspects in one sentence. It can enhance the sentiment classification accuracy, which thus can eventually improve the autonomous decision making ability and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2963927, IEEE Internet of Things Journal

communication ability among object peers in SIoT.

- In MAN, we integrate an attention-over-attention mechanism [12] for reading comprehension into our solution with customized design. Thus, the proposed MAN model can acquire the important information of the sentences and their respective aspect terms while making full use of the mutual information between them.
- The experimental results demonstrate that the performance of our proposed model outperforms several other LSTM-based models. Moreover, we also conduct experiments to demonstrate the effectiveness of our proposed mutual attention mechanism in terms of aspect-level sentiment classification.

The remainder of this paper is organized as follows. Section II describes our problem and discusses the existing challenges. Section III gives some preliminary knowledge. In Section IV, we present the detailed design of our proposed MAN model. Section V conducts some experiments on real-world datasets to show the performance of MAN model. Section VI discusses the related work. In Section VII, we conclude this paper and present prospect for future work.

## II. Problem Scope

In this paper, we aim to develop a new solution to enhance the autonomous decision making ability and communication ability among object peers in SIoT, by relying on the valuable and rich information from social networks. We focus on the sentiment information embedded in social networks contents in particular, and extract such information to capture social users' behaviors or opinions that can serve to SIoT for the intelligent decision making or enhance service needs. Given the fact that each sentence may include multiple aspects instead of one, while their polarities may conflict with each other (i.e., positive polarity and negative polarity), it is necessary to mine multiple aspects information of each sentence so as to capture the correct polarity of this sentence. This contradicts to the existing both document-level and sentence-level methods, where only one aspect is extracted, which is far more than enough to correctly represent one sentence's sentiment information.

In addition to the information of multiple aspect terms, the mutual information between sentences and their respective aspect terms is also important to help recognize the sentiment and the respective polarities. Thus, we will develop new mutual attention mechanism to acquire the important information of sentences and their respective aspect terms while making full use of the mutual information between them for sentiment classification.

Therefore, the goal of this paper is to develop new solution to effectively conduct sentiment classification tasks by taking both aspect terms in each sentence and mutual information between sentence and its aspect terms into consideration to deeply extract the inherent semantic and syntax information of each sentence that can exhibit its sentiment, behaviors, and attitudes. There are two challenges existing in our problem: 1) how to correctly extract the polarity of each sentence in the aspect-level; 2) how to mine the semantic dependence of sentences and their respective aspect terms, and capture the mutual information between them. We propose a novel model in this paper, named mutual attention neural networks (abbreviated as MAN), to conduct the effective sentiment classification tasks by overcoming aforementioned challenges. In particular, a new mutual attention mechanism to extract the mutual information of sentences and their respective aspect terms is developed by us. Then, we employ this proposed mechanism to acquire the mutual relationships of extracted information to complete the aspect-level sentiment classification tasks.

## III. Preliminary Knowledge

### A. Word Embedding

The word embedding is an important technique to expand a word into a vector expression so as to capture the context of a word and relation with other words. Typically, there are two methods to represent words, i.e., discrete representation and distributed representation. One-hot vector is a simple version of discrete representation. That is, assuming there are a total of $n$ words in the lexicon, and for each word, it will be represented by an $n$-dimensional vector, where each word will correspond to one unique position in the $n$-dimensional vector. To express a word with a $n$-dimensional vector, we can set the element in the corresponding position as 1 while all other positions to be 0. This method is simple, but the dimensions of the vector will linearly increase with the number of corpus, resulting in the large space and data sparseness. On the other hand, distributed representation method can address these drawbacks, by training each word to map into a short word vector. This method can create multiple hierarchies or segments where different weights can be assigned to the information displayed by each word, it can be used to study the relationship among words in a common statistical way. The choice of these segments or dimensions can be flexibly determined and each word will be represented by the weight distribution in those segments. For example, "apple" can be expressed as $[0.00, -0.01, 0.03, 0.95]$ while "orange" can be expressed as $[0.01, 0.00, -0.02, 0.97]$. The dimension of a vector can be set to arbitrary value. The GloVe model [13] takes into account both the overall statistics features of the corpus and the local context features, and then introduces the co-occurrence probabilities matrix. It has prominent features of much faster training process and of better performance on small corpora or small vectors, when compared to the traditional word vector training models. Thus, we employ the GloVe model to generate the distributed representation of each word in our method.

### B. LSTM

LSTM is a specific structure of RNN, by adding three control units on the basis of RNN structure, including input gate, forget gate, and output gate. Once information enters into the LSTM, the three units can make adjustment, where information meeting the rules will be kept while the noncompliant information will be forgotten. Therefore, the problem of long sequence dependence in neural networks can be solved based

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2963927, IEEE Internet of Things Journal

on this regulation. In addition, LSTM can avoid vanishing or exploding gradient problems. The general structure of LSTM is shown in Fig. 2.

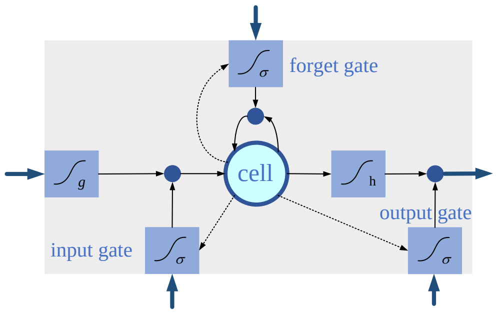



Fig. 2. The basic structure of LSTM.

Given an input word embedding vector $v$ at each time $t$, LSTM is updated as follows:

$$i_t = \sigma(W_i \cdot [h_{t-1}; v_t] + b_i). \tag{1}$$

$$f_t = \sigma(W_f \cdot [h_{t-1}; v_t] + b_f). \tag{2}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}; v_t] + b_o). \tag{3}$$

$$g_t = tanh(W_g \cdot [h_{t-1}; v_t] + b_g). \tag{4}$$

$$c_t = i_t \odot g_t + f_t \odot c_{t-1}. \tag{5}$$

$$h_t = o_t \odot tanh(c_t). \tag{6}$$

where $i$, $f$, and $o$ respectively denote the input gate, forget gate, and output gate, $\sigma$ represents *sigmoid* function, $W_i$, $W_f$, $W_o$, $W_g$ denote weight matrices while $b_i$, $b_f$, $b_o$, $b_g$ denote biases.

## IV. Model Description

We present the detailed design of our proposed MAN model to conduct the sentiment classification tasks in this section. To process the aspect-level sentiment classification tasks, we leverage a prominent attention-over-attention mechanism [12] and develop new solutions with the customized design. In particular, we calculate the average of the two hidden matrices generated by Bi-LSTMs to obtain the initial representations of the sentence and the aspect term, respectively. After multiplying these two matrices, we use two softmax functions to calculate the probability distribution in each column and row of the multiplied matrix. Then, we calculate the average values again to acquire the mutual information between the sentence and the aspect term. After that, we perform the multiplication operation again in the resulted matrices. Fig. 3 shows the flowchart of our proposed MAN model, which includes four modules: 1) the input embedding module: each word is mapped into a low-dimensional vector by using it; 2) Bi-LSTM module: it is employed to learn the hidden semantics of word vectors; 3) the mutual attention module: it is used to acquire the mutual information of the sentence and its aspect terms; 4) the final prediction and model training module: this module is used to perform the final classification of the sentiment polarity.

### A. Constructing Input Embedding Layer

Assume we have one sentence $s = [w_1, w_2, ..., w_i, ..., w_m]$ with the length of $m$ and its aspect term $t = [w_i, w_{i+1}, ..., w_{i+n-1}]$ with the length of $n$ included in the sentence $s$, where $w$ represents the words in $s$. Our goal is to determine the sentiment polarity of sentence $s$ towards aspect term $t$. To represent the words of the aspect terms and that of the sentence, we aim to mapped into the low-dimensional vectors with real values via the distribution representation word embedding method [14]. The prominent distribution representation embedding method, i.e., GloVe [13], is employed to generate an embedding matrix, denoted as $L \in R^{|V| \times d_l}$, serving as the word vector lookup table, where $d_l$ and $\mid V \mid$ represent the embedding dimension and vocabulary size, respectively. We query the vector lookup table to find the respective low-dimensional vector for each word. For the words that are not found in the embedding matrix $L$, the corresponding values will be set to zero. Thus, in the end, we have two sets of word vectors $S = [v_1; v_2; ...; v_i; ...; v_m]$, $S \in R^{m \times d_l}$, and $T = [v_i; v_{i+1}; ...; v_{i+n-1}]$, $T \in R^{n \times d_l}$, corresponding to the sentence and its aspect term, respectively.

### B. Obtaining Semantic Dependence using Bi-LSTM

After obtaining the word vectors $S$ and $T$, the hidden semantics of $S$ and $T$ can be learned respectively by using two Bi-LSTMs. For each Bi-LSTM, it consists of two LSTMs, where one is a forward LSTM and the other one is a backward LSTM. By using Bi-LSTM, we can capture the bidirectional semantic dependence of the sentence, thereby avoiding the inability to encode a sequence of information from back to front if only LSTM is employed. The structure of the Bi-LSTM module in our model is illustrated in Fig. 4.

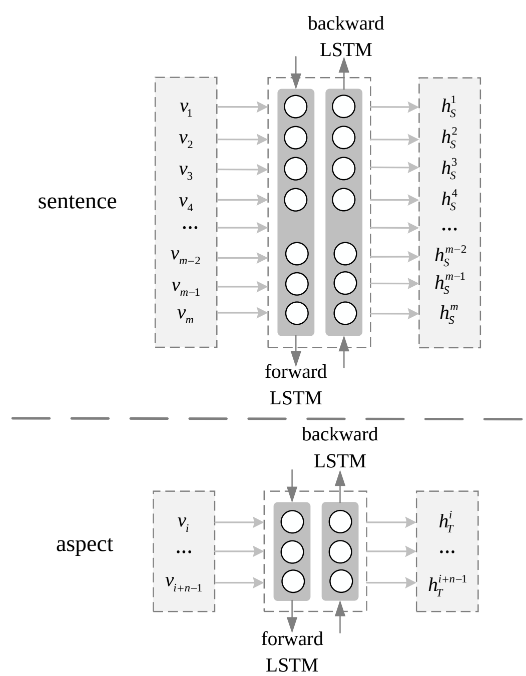



Fig. 4. The structure of the Bi-LSTM module.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2963927, IEEE Internet of Things Journal

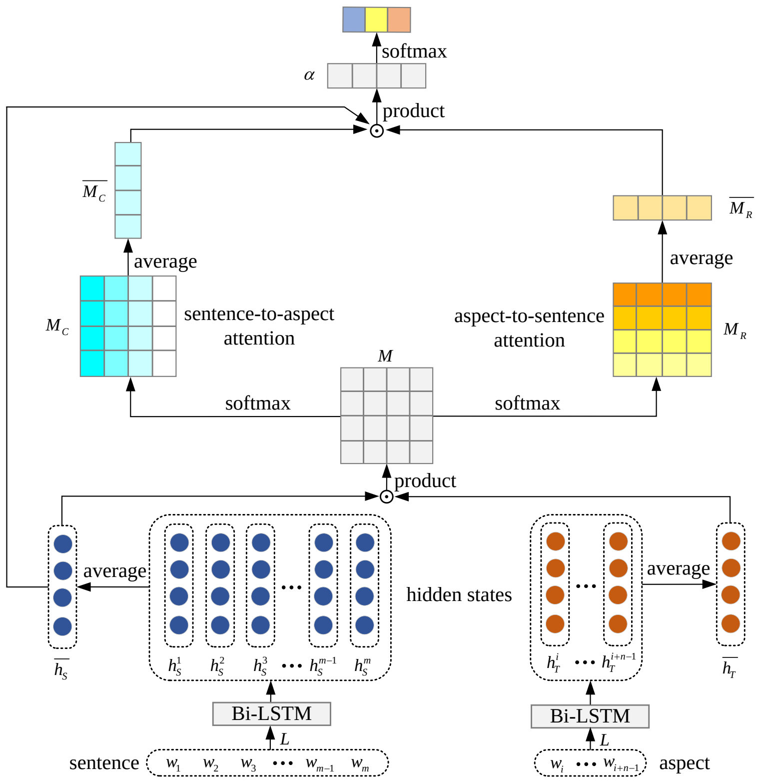



Fig. 3. Model structure, containing input embedding module, Bi-LSTM module, mutual attention module, and the final prediction and model training module.

One forward LSTM and one backward LSTM constitute a Bi-LSTM. Each word in the sentence is represented by connecting the forward hidden states and the backward ones. For forward LSTM, given an input $S$, we can obtain a sequence of hidden states $\overrightarrow{h_S} \in R^{m \times d_h}$, where $d_h$ represents the dimension of hidden states. For backward LSTM, given the input $S$, we can obtain another sequence of hidden states $\overleftarrow{h_S} \in R^{m \times d_h}$. The final output hidden states $h_S = [h_S^1; h_S^2; ...; h_S^i; ...; h_S^m]$, $h_S \in R^{m \times 2d_h}$ are generated by connecting the hidden states of the forward and backward LSTMs. The equations are expressed as follows:

$$\overrightarrow{h_S} = \overrightarrow{LSTM}(S). \tag{7}$$

$$\overleftarrow{h_S} = \overleftarrow{LSTM}(S). \tag{8}$$

$$h_S = [\overrightarrow{h_S}; \overleftarrow{h_S}]. \tag{9}$$

Using the same method, we can also obtain a sequence of hidden states $h_T = [h_T^i; h_T^{i+1}; ...; h_T^{i+n-1}]$, $h_T \in R^{n \times 2d_h}$ when given an input $T$. The equations are expressed as follows:

$$\overrightarrow{h_T} = \overrightarrow{LSTM}(T). \tag{10}$$

$$\overleftarrow{h_T} = \overleftarrow{LSTM}(T). \tag{11}$$

$$h_t = [\overrightarrow{h_T}; \overleftarrow{h_T}]. \tag{12}$$

### C. Capturing Mutual Attention

After obtaining the hidden semantic representations of the sentence $h_S$ and its aspect term $h_T$, which are generated by Bi-LSTMs, we first calculate the average of the two hidden matrices and obtain $\overline{h_S}$ and $\overline{h_T}$, $\overline{h_S}$ and $\overline{h_T}$ are used as the the initial representations of the sentence and its aspect term:

$$\overline{h_S} = \sum_{i=1}^{m} \frac{h_S^i}{m}. \tag{13}$$

$$\overline{h_T} = \sum_{i=1}^{n} \frac{h_T^i}{n}. \tag{14}$$

Second, we multiply $\overline{h_S}$ and $\overline{h_T}$, obtaining a sentence-aspect matching matrix $M$, that indicates the matching degree between the words of the sentence and its aspect term:

$$M = \overline{h_S} \cdot \overline{h_T}^T. \tag{15}$$

Third, we use two softmax functions to calculate the probability distribution in each column and row of the matching matrix $M$. The results are called the sentence-to-aspect attention matrix $M_C$ and the aspect-to-sentence attention matrix $M_R$:

$$M_{C_{ij}} = \frac{exp(M_{ij})}{\sum_i exp(M_{ij})}. \tag{16}$$

$$M_{R_{ij}} = \frac{exp(M_{ij})}{\sum_j exp(M_{ij})}. \tag{17}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2963927, IEEE Internet of Things Journal

At this point, we have obtained both sentence-to-aspect attention matrix $M_C$ and aspect-to-sentence attention matrix $M_R$. We calculate the average of $M_C$ and $M_R$ to acquire the mutual information between the sentence and its aspect term:

$$\overline{M_C} = \frac{\sum_j M_{C_{ij}}}{2d_h}. \tag{18}$$

$$\overline{M_R} = \frac{\sum_i M_{R_{ij}}}{2d_h}. \tag{19}$$

Then, we multiply $\overline{M_C}$ by $\overline{M_R}$ and get the results denoted as $p$, representing the final sentence-level attention, i.e.,

$$p = \overline{M_C}^T \cdot \overline{M_R}^T. \tag{20}$$

Ultimately, the final sentence representation $\alpha$ is obtained by multiplying $\overline{h_s}$ and $p$, we have:

$$\alpha = \overline{h_S} \cdot p. \tag{21}$$

### D. Training Model and Finally Predicting

The final sentence representation $\alpha$ is regarded as the final classification feature, we plan to project $\alpha$ into the targeted $C$ classes space by feeding $\alpha$ into a linear layer. The equation can be expressed as follows:

$$x = W_l \cdot \alpha + b_l. \tag{22}$$

where $b_l$ is the bias and $W_l$ is the weight matrix. Then, to calculate the probability of the sentence $s$ with sentiment polarity $c \in C$, we feed $x$ into a softmax layer, i.e.,

$$P_c = \frac{exp(x_c)}{\sum_{i \in C} exp(x_i)}. \tag{23}$$

where $P_c$ is the predicted probability of sentiment class $c$. The label whose probability is the highest will be considered as the ultimate predicted sentiment polarity. We train the model by minimizing the cross-entropy with $L_2$ regularization:

$$Loss = -\sum_{(s,t)\in D}\sum_{c\in C} P_c^g(s,t) \cdot log(P_c(s,t)) + \lambda \left\|\theta\right\|^2. \tag{24}$$

where $(s,t)$ is a sentence-aspect pair, $D$ is the collection of training data, $P_c(s,t)$ is the probability that our model predicts that $(s,t)$ belongs to class $c$, $P_c^g(s,t)$ is an indicator function with value 0 or 1, $\theta$ is the weight matrix in the LSTM and linear layer, and $\lambda$ is the $L_2$ regularization parameter.

## V. Experimental

To evaluate the performance of our proposed MAN model in the processing of aspect-level sentiment classification tasks, we conduct experiments on the real-world datasets in this section. The detailed experiments setting will be listed, including the distribution of the datasets, the setting of the hyperparameters, and the detailed case study.

### A. Experimental Setting

To evaluate the performance of our MAN model, we take three real-world datasets: LAPTOP, REST, and TWITTER, which have been widely used in previous research, and do experiments on them. The detailed description of these three datasets is given: LAPTOP and REST are from the SemEval ABSA challenge [15], containing reviews in the laptop and restaurant domains, respectively, collected from social media platforms. It should be noted that they not only include the positive, negative and neutral these three kinds of sentiment polarities, but also contain the fourth category, which is conflict, it represents that a sentence expresses both positive and negative opinions on one aspect. As the number of the reviews for the fourth category is relatively small, it makes the entire dataset unbalance. Thus, the conflict category is removed [16]. In our experiment, we only use the LAPTOP and REST datasets that include positive, negative and neutral sentiment polarities. TWITTER includes a set of twitter posts, which are collected by Dong *et al.* [17]. Notably, all reviews in these three datasets are tagged with three sentiment polarities: negative, neutral, and positive. The datasets that we have used in the experiments have the unified format, where each sentence has a list including aspects and the sentiment polarities information [15]. An example of the unified format is shown in Fig. 5.


```
<sentence id = “11351725#582163#9” >
    <text>Our waiter was friendly and it is a shame that he didnt have a supportive staff to work with.</text>
    <aspectTerms>
        <aspectTerm term= “waiter” polarity= “positive” form= “4” to= “10” />
        <aspectTerm term= “staff” polarity= “negative” form= “74” to= “79” />
    </aspectTerms>
    <aspectCategories>
        <aspectCategory category= “service” polarity= “conflict” />
    </aspectCategories>
</sentence>
```


Fig. 5. The unified format of the datasets.

Our goal here is to identify the corresponding polarities of the aspects in the review. The numbers of test and training instances of each category in the three datasets is shown in Table I.

TABLE I
DISTRIBUTIONS OF THE DATASETS FROM SEMEVAL-2014 TASK 4 AND TWITTER. THE NUMBERS IN THE TABLE INDICATE THE NUMBER OF SENTENCE-ASPECT PAIRS.

| Datasets | | Positive | Negative | Neutral |
|---|---|---|---|---|
| REST | Train | 2164 | 807 | 637 |
| | Test | 728 | 196 | 196 |
| LAPTOP | Train | 994 | 870 | 464 |
| | Test | 341 | 128 | 169 |
| TWITTER | Train | 1561 | 1560 | 3127 |
| | Test | 173 | 173 | 346 |

In our experiment, 300-dimensional GloVe vectors are used to initialize the word embedding of sentences and aspect terms in the datasets. For the words that are not found in the embedding matrix $L$, the corresponding values will be set to zero. The relatively short sentences are zero-padded until the length is the same as the length of the longest sentence in the datasets. By sampling a uniform distribution $U(-10^{-4}, 10^{-4})$, all values of the weight matrix are initialized, and all biases are initialized to 0. The $L_2$ regularization coefficient is set to be $10^{-5}$ and the dropout rate [18] is set to be 0.3. We choose Adam optimizer [19] to minimize the cross-entropy loss with the learning rate initialized to be 0.001, and the batch size set to be 32. We adopt two metrics to evaluate the classification performance of our model: Accuracy and Macro-F1 score, where the first one is usually adopted to standard classification problems and the second one is more suitable for multi-label classification tasks. They can be calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} = \frac{TP + TN}{N}. \tag{25}$$

$$MacroPrecision = \frac{1}{|C|} \sum_{i=1}^{|C|} \frac{TP_i}{TP_i + FN_i}. \tag{26}$$

$$MacroRecall = \frac{1}{|C|} \sum_{i=1}^{|C|} \frac{TP_i}{TP_i + FP_i}. \tag{27}$$

$$Macro - F1 = \frac{2 \times MacroPrecision \times MacroRecall}{MacroPrecision + MacroRecall}. \tag{28}$$

where $N$ denotes the total number of testing samples, $|C|$ is the amount of classes. $TP$, $TN$, $FP$, $FN$ are described in Table II.

TABLE II
CLASSIFICATION RESULT CONFUSION MATRIX.

| | | True Label | |
|---|---|---|---|
| | | Yes | No |
| Classifier | Yes | TP (True Positive) | FP (False Positive) |
| Label | No | FN (False Negative) | TN (True Negative) |

## B. Compared Solutions

We plan to compare the performance of our MAN model with the the following existing models:

- **Majority**: This method distributes most of the sentiment labels of the training set on each sample in the test set. It is a basic baseline method.
- **LSTM**: This method uses only one LSTM to model the sentence, and the ultimate averaged hidden vector is regarded as the representation of the sentence for the final classification.
- **AE-LSTM** and **ATAE-LSTM** [20]: They use the embeddings of aspect terms to generate attention vectors, thus focusing on different parts of the sentence. ATAE-LSTM combines the aspect embedding and the word embedding vector to represent the context and generate attention vectors.
- **TD-LSTM** [21]: It uses two LSTMs, namely, one forward LSTM and one backward LSTM, aiming to model the front part and the left part of the aspect term separately. The final classification is based on concatenated context representations.
- **MemNet** [22]: This method uses multi-hop attention mechanism, the attention is used many times for word embedding instead of combing the results of different attentions, and only the last attention output is input into the softmax function for prediction.
- **IAN** [23]: The author uses the hidden states from the context and the target to generate attention vectors for the target and the corresponding context. The final classification is based on the concatenation of the context representation and the target representation.
- **AOA** [24]: It also applies the attention-over-attention mechanism [12], which was proposed for reading comprehension tasks. Different from our model, the hidden matrices of the sentence and the aspect term are directly multiplied without calculating the average of them.
- **BILSTM-ATT-G** [25]: In this method, attention-based LSTM is adopted to model the front part and the left part around the target separately and control the value of the front part, the left part and the whole sentence for prediction by using gates.

## C. Performance Comparisons

For different random initializations, the performance of the model fluctuates, which is a common initialization problem in the process of neural network training [26]. We have conducted multiple experiments on the three datasets to test the performance of our MAN model and found the fluctuation range is relative small. For example, on LAPTOP dataset, the classification Accuracy values fluctuates from $73.35\%$ to $75.71\%$ and the Macro-F1 score values fluctuates from $70.87\%$ to $72.89\%$. Similarly, on the other two datasets, i.e., REST and TWITTER, the Accuracy values and the Macro-F1 score values also fluctuate within a relatively small range. To show the performance of our MAN model, we run experiments 10 times and calculate the average values of Accuracy and Macro-F1 score on each dataset. The comparison of our MAN model with other respective models as listed in Section V-B is shown in Table III. From this table, we can see our proposed MAN model always outperforms other respective models in terms of Accuracy and Macro-F1 score. The reason is that, the first two methods, i.e., Majority and LSTM, are designed to classify the sentiment polarity at the text level, they can judge the sentiment polarity of the whole sentence and cannot capture the information of the aspect terms, so their performances are the worst. AE-LSTM and ATAE-LSTM generate attention vectors by using the embeddings of the aspect terms, they

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2963927, IEEE Internet of Things Journal

TABLE III
THE COMPARISON RESULTS (%). CLASSIFICATION PERFORMANCE OF DIFFERENT METHODS ON SEMEVAL-2014 TASK 4 AND TWITTER DATASETS. WE RUN THE MAN MODEL 10 TIMES, AND SHOW THE MEAN OF THE ACCURACY AND THE MEAN OF THE MACRO-F1 SCORE. THE BEST PERFORMANCES ARE IN BOLD.

| Methods | REST | | LAPTOP | | TWITTER | |
|---|---|---|---|---|---|---|
| | Accuracy | Macro-F1 | Accuracy | Macro-F1 | Accuracy | Macro-F1 |
| Majority | 53.50 | 33.33 | 65.00 | 33.33 | 50.00 | 33.33 |
| LSTM | 75.45 | 66.57 | 66.77 | 58.62 | 64.16 | 61.92 |
| AE-LSTM | 76.51 | 64.19 | 68.74 | 65.35 | 68.47 | 66.70 |
| ATAE-LSTM | 77.23 | 66.69 | 68.50 | 63.81 | 68.35 | 65.06 |
| TD-LSTM | 78.04 | 66.24 | 69.44 | 63.24 | 70.66 | 68.18 |
| MemNet | 78.75 | 66.74 | 70.53 | 63.70 | 70.23 | 68.11 |
| IAN | 78.57 | 67.94 | 71.78 | 66.26 | 69.51 | 67.88 |
| AOA | 79.98 | 68.96 | 73.07 | 68.52 | 70.95 | 67.88 |
| BILSTM-ATT-G | 79.75 | 69.12 | 73.11 | 69.97 | 70.86 | 68.63 |
| MAN | **80.71** | **70.95** | **74.13** | **71.93** | **72.12** | **70.13** |

TABLE IV
THE PERFORMANCE OF ABLATED MAN (%). CLASSIFICATION PERFORMANCE OF MAN-R, MAN-C, AND MAN ON SEMEVAL-2014 TASK 4 AND TWITTER DATASETS. WE RUN THEM 10 TIMES, AND SHOW THE MEAN OF THE ACCURACY AND THE MEAN OF THE MACRO-F1 SCORE. THE BEST PERFORMANCES ARE IN BOLD.

| Methods | REST | | LAPTOP | | TWITTER | |
|---|---|---|---|---|---|---|
| | Accuracy | Macro-F1 | Accuracy | Macro-F1 | Accuracy | Macro-F1 |
| MAN-R | 79.01 | 68.44 | 72.12 | 68.72 | 70.76 | 69.12 |
| MAN-C | 78.98 | 67.63 | 71.43 | 67.74 | 69.75 | 68.56 |
| MAN | **80.71** | **70.95** | **74.13** | **71.93** | **72.12** | **70.13** |

perform better as the attention mechanism can help the model focus on the target word in one sentence. In TD-LSTM, target word is took into consideration, and two LSTMs are used to model the front part and the left part of the target separately. It performs better than ATAE-LSTM but still worse than MemNet, as the attention mechanism is not adopted. MemNet uses attention many times for word embedding, and the last attention output is used for prediction. IAN uses the hidden states from the context and the target to generate attention vectors for the target and the context, respectively. AOA applies an attention-over-attention model to enhance the classification accuracy. BILSTM-ATT-G uses attention-based LSTM to model the front part and the left part around the target separately. Although the above models use attention mechanisms in many ways, their accuracy values are lower than that of ours. The reason can be concluded as following: first, after obtaining the hidden semantic representations of the sentence and its aspect term that are generated by Bi-LSTMs, we calculate the average to obtain their initial representations; second, we use the mutual attention mechanism to acquire the important information of the sentence and its aspect terms while making full use of the mutual information between them. After adjusting the parameters many times, we obtained the better results than several other LSTM-based models.

### D. Time Cost Comparisons

We consider the same 300-dimensional GloVe word vectors and implement several typical LSTM-based models (i.e., LSTM, MemNet, TD-LSTM, ATAE-LSTM, IAN, and AOA) on SemEval-2014 Task 4 and TWITTER datasets, and run them on the machine with the configuration of Windows 10 system with 2×Intel(R) Xeon(R) Silver 4214 CPU, GeForce RTX 2080 Ti GPU, and 128GB RAM. We run 10 times for each of the compared solutions and our MAN model. The averaged computational cost of each method is shown in Table V.

From these results, we can see the computational cost of our MAN model is higher than the compared solutions. The reason is that our MAN model needs to use the mutual attention mechanism to acquire the mutual information between sentences and their respective aspect terms, which is inevitably bringing more computation complexity. However,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2963927, IEEE Internet of Things Journal

TABLE V
THE AVERAGED RUNTIME (SECONDS) OF LSTM, MEMNET, TD-LSTM, ATAE-LSTM, IAN, AOA, AND OUR MAN MODEL.

| Methods | Time Cost | | |
|---|---|---|---|
| | REST | LAPTOP | TWITTER |
| LSTM | 64 | 44 | 85 |
| MemNet | 88 | 59 | 139 |
| TD-LSTM | 97 | 66 | 144 |
| ATAE-LSTM | 121 | 81 | 177 |
| IAN | 141 | 94 | 216 |
| AOA | 157 | 97 | 228 |
| MAN | 164 | 115 | 237 |

from Table III, we can see the classification accuracy of our MAN model outperforms other compared solutions.

### *E. Performance of Ablated MAN*

In this section, to show the impact of each component on our MAN model, we introduce two schemes: MAN-R and MAN-C. The mutual attention mechanism structures of MAN-R and MAN-C are shown in Fig. 6.

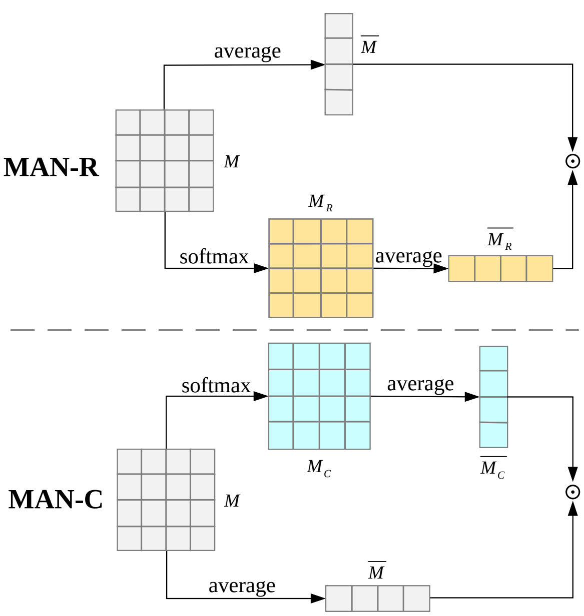



Fig. 6. The mutual attention mechanism structures of MAN-R and MAN-C.

**MAN-R**. After multiplying $\overline{h_S}$ and $\overline{h_T}$, and obtaining the sentence-aspect matching matrix $M$, we only use one softmax function to calculate the probability distribution in each row of matrix $M$, and the result is matrix $M_R$. We calculate the average of $M$ and $M_R$, then multiply $\overline{M}$ and $\overline{M_R}$, the result is called $p_R$. Then, the final sentence representation $\alpha_R$ is obtained by multiplying $\overline{h_S}$ and $p_R$. The equations are summarized as follows:

$$\overline{M} = \frac{\sum_j M_{ij}}{2d_h}. \tag{29}$$

$$M_{R_{ij}} = \frac{exp(M_{ij})}{\sum_j exp(M_{ij})}. \tag{30}$$

$$\overline{M_R} = \frac{\sum_i M_{R_{ij}}}{2d_h}. \tag{31}$$

$$p_R = \overline{M}^T \cdot \overline{M_R}^T. \tag{32}$$

$$\alpha_R = \overline{h_S} \cdot p_R. \tag{33}$$

**MAN-C**. Different from MAN-R, in this scheme, softmax function is used to calculate the probability distribution in each column of matrix $M$, and the result is matrix $M_C$. We then calculate the average of $M$ and $M_C$, multiply $\overline{M}$ and $\overline{M_C}$, to obtain the result, denoted as $p_C$. The final sentence representation $\alpha_C$ is obtained by multiplying $\overline{h_S}$ and $p_C$. The equations are summarized as follows:

$$\overline{M} = \frac{\sum_i M_{ij}}{2d_h}. \tag{34}$$

$$M_{C_{ij}} = \frac{exp(M_{ij})}{\sum_i exp(M_{ij})}. \tag{35}$$

$$\overline{M_C} = \frac{\sum_j M_{C_{ij}}}{2d_h}. \tag{36}$$

$$p_C = \overline{M_C}^T \cdot \overline{M}^T. \tag{37}$$

$$\alpha_C = \overline{h_S} \cdot p_C. \tag{38}$$

We feed $\alpha_R$ and $\alpha_C$ into softmax layer respectively to calculate the probability with sentiment polarity. We run MAN-R, MAN-C, and MAN on SemEval-2014 Task 4 and TWITTER datasets 10 times, the means of both Accuracy and the Macro-F1 score are presented in Table IV.

From Table IV, we can see that the processes of calculating aspect-to-sentence attention and sentence-to-aspect attention are very important in the mutual attention mechanism for the sentiment classification. On the other hand, we can see the Accuracy and the Macro-F1 score of MAN-C are both lower than that of MAN-R, which shows that the aspect-to-sentence attention is more important than the sentence-to-aspect attention. Our MAN model includes both aspect-to-sentence attention and sentence-to-aspect attention, thus, it can outperform both MAN-R and MAN-C.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2963927, IEEE Internet of Things Journal

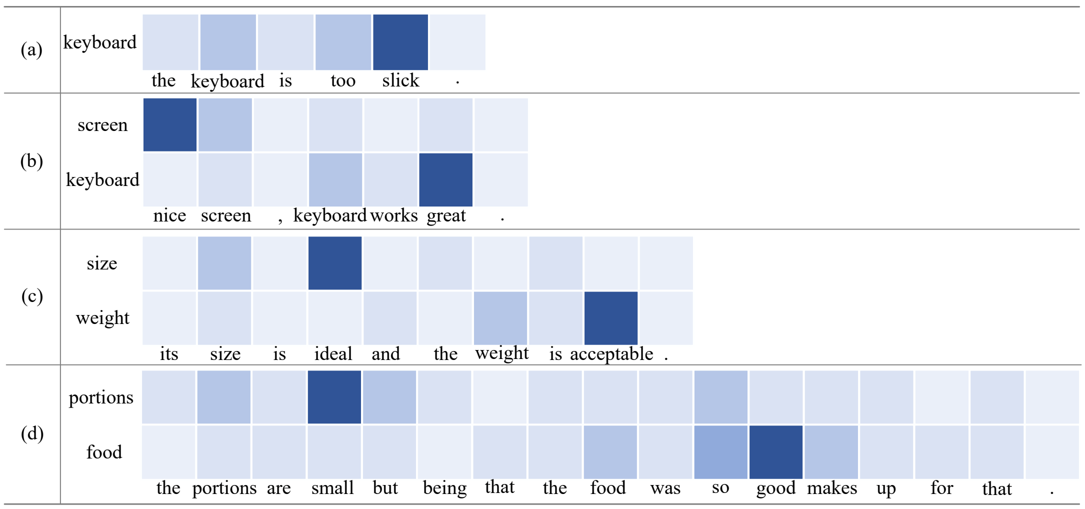



Fig. 7. Attention weight visualization of the sentence and the aspect terms.

| | Sentence | True | LSTM | ATAE-LSTM | BILSTM-ATT-G | MAN-R | MAN-C | MAN |
|---|---|---|---|---|---|---|---|---|
| (e) | Service was good and so was the atmosphere . | (↑,↑) | (↑,↑) | (↑,↑) | (↑,↑) | (↑,↑) | (↑,↑) | (↑,↑) |
| (f) | The staff should a bit more friendly. | ↓ | ↑ | ↑ | ↑ | ↑ | ↑ | ↓ |
| (g) | I need graphic power to run my Adobe Creative apps efficiently. | (→,→) | (↑,↑) | (→,↑) | (→,→) | (→,→) | (→,→) | (→,→) |
| (h) | I wish the volume could be louder and the mouse didnt break after only a month. | (↓,↓) | (↑,↑) | (↑,↑) | (↑,↓) | (↑,↓) | (↑,↑) | (↓,↓) |
| (i) | Dessert was also to die for! | ↑ | ↓ | ↓ | ↑ | ↑ | ↑ | ↑ |

Fig. 8. Examples of classification results. The notations ↑, ↓, and → indicate positive, negative, and neutral respectively.

### *F. Case Study*

In this section, we first cite four reviews from the three datasets for the case study to test the validity of our MAN model: "The keyboard is too slick.", with one aspect term "keyboard"; "Nice screen, keyboard works great.", with two aspect terms "screen" and "keyboard"; "It's size is ideal and the weight is acceptable", with two aspect terms "size" and "weight"; "The portions are small but being that the food was so good makes up for that.", with two aspect terms "portions" and "food". We use MAN to model these four sentences and their respective aspect terms, and visualize the attention weights, as shown in Fig. 7. The depth of the blue color indicates how important a word is to the entire sentence. The sentiment polarity indicating words of aspects in the four sentences can be automatically pointed out in the sentence by a colored mark. Therefore, we can obtain that the attention mechanism can find the relatively important word in the whole sentence. In sentence (a), for aspect "keyboard", the relatively important word is "slick"; in sentence (b), for aspect "screen", the relatively important word is "nice", and for aspect "keyboard", the relatively important word is "great"; in sentence (c), for aspect "size", the relatively important word is "ideal", and for aspect "weight", the relatively important word is "acceptable"; in sentence (d), for aspect "portions", the relatively important word is "small", and for aspect "food", the relatively important word is "good".

Furthermore, we also list some sentences to further show the superiority of our model. We can obtain the examples of classification results shown in Fig. 8. As shown, two aspects are given in sentence (e): for aspect "service", the sentiment polarity is positive, and for aspect "atmosphere", the sentiment polarity is also positive. These six methods can distinguish the sentiment polarity of the aspect accurately. In sentence (f), the sentiment polarity according to aspect "staff" is negative, the grammar is so complicated that except for the MAN model, other methods cannot judge the sentiment polarity correctly, it should be the word "friendly" that affects the accuracy of these four methods. In sentence (g), the sentiment polarities of the two aspects "graphic power" and "Adobe Creative apps" are all neutral, in addition to LSTM and ATAE-LSTM, other methods can correctly distinguish the sentiment polarities. Similar to sentence (f), the grammar of sentence (h) is also very complicated, the sentiment polarities of the two aspects "volume" and "mouse" are all negative, and only the MAN model can correctly classify the polarities. In sentence (i), except for LSTM and ATAE-LSTM models, other four methods can judge the sentiment polarities correctly, it should

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2963927, IEEE Internet of Things Journal

be the word "die" that affects the accuracy of LSTM and ATAE-LSTM.

Next, we focus on analyzing the three models proposed in this paper. Through the MAN model, we can accurately determine the sentiment polarities of the two aspects. But through the MAN-R and the MAN-C model, we can't get the totally correct classification results. Table IV shows that the aspect-to-sentence attention is more important than the sentence-to-aspect attention, the accuracy of MAN-C is lower than that of MAN-R. And the processes of calculating aspect-to-sentence attention and sentence-to-aspect attention are both very important in the mutual attention mechanism for the sentiment classification. The MAN model includes both aspect-to-sentence attention and sentence-to-aspect attention, thus, it can outperform both MAN-R and MAN-C. In addition to these sentences, we also cite other sentences in the datasets for case study, and conclude that the classification accuracy of MAN-R and MAN-C is lower than that of MAN when the grammar is complicated, and the accuracy of MAN-C is lower than that of MAN-R.

As can be learned from the above examples, our model can work well when performing aspect-level sentiment classification tasks.

## VI. Related Work

Aspect-level sentiment classification aims to judge the sentiment polarity of a particular aspect in one sentence. The research methods may mainly be divided into two kinds: traditional machine learning methods and neural networks methods. On one hand, the main idea of the traditional machine learning method is to design a set of features such as bag-of-words and sentiment lexicon to train a classifier like support vector machine (SVM) for aspect-level sentiment classification. For example, Nasukawa *et al.* [27] first parse the sentences and then use predefined rules to classify the aspect terms; Jiang *et al.* [28] create several feature-related features based on the grammatical structure of sentences, thus improving the target-dependent sentiment classification.

On the other hand, a number of neural-network-based models have been proposed, which use recurrent neural networks [29], recursive neural networks [30], and CNNs [11]. Neural-network-based methods automatically learn feature representations without the need for intensive feature engineering, and they have made great achievements in sentiment classification tasks. Based on LSTM, the most typical classification methods are created. There are two advantages of LSTM: first, it can solve the problem of long sequence dependence in neural networks; second, it can avoid vanishing or exploding gradient problems [10]. Tang *et al.* [21] develop two LSTM-based neural networks, which are called target-dependent long short-term memory (TD-LSTM) networks and target-connection long short-term memory (TC-LSTM) networks, to model the context of target words and solve the classification problem. These two methods predict the sentiment polarity by using the last hidden states of the two LSTMs.

Although the neural-network-based models improve the classification accuracy to a certain extent, when the sentiment words are farther away from the modified aspect terms, these models cannot effectively capture long-distance information. Hence, to improve the accuracy of the overall sentiment classification, the attention mechanism is applied. After achieving excellent results in the image processing field, researchers began to study how to introduce the attention mechanism into natural language processing tasks. Dzmitry *et al.* [31] are the first to propose the soft attention model and apply it to the machine translation field. Alexander *et al.* [32] use an attention mechanism to summarize the abstract of one sentence. Sukhbaatar *et al.* [33] apply the attention mechanism to question-answering tasks. After the attention mechanism achieved good results in the above areas, it was also applied to the sentiment classification tasks [34]. For example, Wang *et al.* [20] generate attention vectors by using the aspect embedding, thereby focusing attention on different parts of one sentence; Cheng *et al.* [35] propose a hierarchical attention (HEAT) network to solve the problem of whether an unconcerned sentiment word is meaningful for the specified aspect semantically; Ma *et al.* [23] use the hidden states from the context and the aspect terms to generate attention vectors for the target and the corresponding context; Li *et al.* [36] use a CNN to extract the most important classification features of the sentence; Duan *et al.* [37] develop a reinforcement-learning-based method that automatically introduces target-specific sentence representations into the tree structure; Liu *et al.* [25] use gates to control the value of the front part, the left part and the whole sentence for prediction. Compared with these methods, our MAN model is effective which can acquire the important information of the sentences and their respective aspect terms and simultaneously take full advantage of the mutual information between them for sentiment classification.

## VII. Conclusion and Future Work

Nowadays, the combination of SIoT and artificial intelligence shows a promising development prospect with the rapid development of artificial intelligence technology. The analysis of the data based on the opinions expressed by users can improve the autonomous decision making ability and communication ability among object peers of SIoT. We have proposed a new model in this paper, named mutual attention neural networks (abbreviated as MAN), to effectively mine the affluent information of sentences and conduct the aspect-level sentiment classification. Our MAN model can acquire the important information of the sentences and their respective aspect terms while taking full advantage of the mutual information between them. When given different aspects, it can focus on different parts of one sentence. The experiments on the three real-world datasets: LAPTOP, REST, and TWITTER, have demonstrated the validity and competitiveness of this model and that it can improve the performance of sentiment classification when compared with multiple baselines.

Although the method we proposed has achieved satisfactory results in solving the sentiment classification problem, in subsequent work, we intend to add more grammatical elements to our model on the basis of the attention mechanism to understand the meaning of the sentence in more depth, enhance the sentiment classification performance, and bring better experiences to users.

## REFERENCES

[1] K. Ashton *et al.*, "That internet of things thing," *RFID journal*, vol. 22, no. 7, pp. 97–114, 2009.

[2] J. Scott, "Social network analysis," *Sociology*, vol. 22, no. 1, pp. 109–127, 1988.

[3] L. Atzori, A. Iera, G. Morabito, and M. Nitti, "The social internet of things (siot)–when social networks meet the internet of things: Concept, architecture and network characterization," *Computer networks*, vol. 56, no. 16, pp. 3594–3608, 2012.

[4] C. D. Manning, C. D. Manning, and H. Schütze, *Foundations of statistical natural language processing*. MIT press, 1999.

[5] B. Liu, "Sentiment analysis and opinion mining," *Synthesis lectures on human language technologies*, vol. 5, no. 1, pp. 1–167, 2012.

[6] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and Trends in Information Retrieval*, vol. 2, no. 1, pp. 1–135, 2008.

[7] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, 2002, pp. 79–86.

[8] P. D. Turney, "Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 417–424.

[9] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[11] Y. Kim, "Convolutional neural networks for sentence classification," *empirical methods in natural language processing*, pp. 1746–1751, 2014.

[12] Y. Cui, Z. Chen, S. Wei, S. Wang, T. Liu, and G. Hu, "Attention-over-attention neural networks for reading comprehension," *meeting of the association for computational linguistics*, vol. 1, pp. 593–602, 2017.

[13] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[14] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of machine learning research*, vol. 3, no. Feb, pp. 1137–1155, 2003.

[15] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, "Semeval-2014 task 4: Aspect based sentiment analysis," pp. 27–35, 2014.

[16] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent attention network on memory for aspect sentiment analysis," in *Proceedings of the 2017 conference on empirical methods in natural language processing*, 2017, pp. 452–461.

[17] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "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)*, vol. 2, 2014, pp. 49–54.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *international conference on learning representations*, 2015.

[20] Y. Wang, M. Huang, L. Zhao *et al.*, "Attention-based lstm for aspect-level sentiment classification," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, 2016, pp. 606–615.

[21] D. Tang, B. Qin, X. Feng, and T. Liu, "Effective lstms for target-dependent sentiment classification," *international conference on computational linguistics*, pp. 3298–3307, 2016.

[22] D. Tang, B. Qin, and T. Liu, "Aspect level sentiment classification with deep memory network," *empirical methods in natural language processing*, pp. 214–224, 2016.

[23] D. Ma, S. Li, X. Zhang, and H. Wang, "Interactive attention networks for aspect-level sentiment classification," *international joint conference on artificial intelligence*, pp. 4068–4074, 2017.

[24] B. Huang, Y. Ou, and K. M. Carley, "Aspect level sentiment classification with attention-over-attention neural networks," in *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*. Springer, 2018, pp. 197–206.

[25] J. Liu and Y. Zhang, "Attention modeling for targeted sentiment," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, vol. 2, 2017, pp. 572–577.

[26] I. Sutskever, J. Martens, G. E. Dahl, and G. E. Hinton, "On the importance of initialization and momentum in deep learning," pp. 1139–1147, 2013.

[27] T. Nasukawa and J. Yi, "Sentiment analysis: Capturing favorability using natural language processing," in *Proceedings of the 2nd international conference on Knowledge capture*. ACM, 2003, pp. 70–77.

[28] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao, "Target-dependent twitter sentiment classification," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 151–160.

[29] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[30] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.

[31] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *international conference on learning representations*, 2015.

[32] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," *empirical methods in natural language processing*, pp. 379–389, 2015.

[33] S. Sukhbaatar, J. Weston, R. Fergus *et al.*, "End-to-end memory networks," in *Advances in neural information processing systems*, 2015, pp. 2440–2448.

[34] K. Cho, B. Van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," *empirical methods in natural language processing*, pp. 1724–1734, 2014.

[35] J. Cheng, S. Zhao, J. Zhang, I. King, X. Zhang, and H. Wang, "Aspect-level sentiment classification with heat (hierarchical attention) network," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 2017, pp. 97–106.

[36] X. Li, L. Bing, W. Lam, and B. Shi, "Transformation networks for target-oriented sentiment classification," *meeting of the association for computational linguistics*, vol. 1, pp. 946–956, 2018.

[37] J. Duan, X. Ding, and T. Liu, "Learning sentence representations over tree structures for target-dependent classification," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, vol. 1, 2018, pp. 551–560.

**Nan Jiang** was born in 1981. He received his PhD degree (2008) from the Nanjing University of Aeronautics and Astronautics. From 2013 to 2014, he was a research scholar in the Complex Networks and Security Research Lab at Virginia Tech, Blacksburg, VA, USA. Currently, he is a professor at Department of Internet of Things and the director of the Intelligent Sensor Networks Lab at East China Jiaotong University. He received several National Science Foundation of China (NSFC) Grants, the Career Award for Young Scientists of Jiangxi Province of China and leaders of academic and technical of Jiangxi Province of China. He has published more than 60 papers in international journals and conferences and served as program chairs and committee for many international conferences such as CSS 2019, IEEE EUC 2017, ISICA 2015 etc. He is a professional member of the ACM, a professional member of the IEEE, and a senior member of the China Computer Federation (CCF).

His current research interests lie in the Internet of Things, Cyber Physical Systems, and Social Networks.

**Fang Tian** was born in 1995. She received her BS degree in Internet of Things Engineering from East China Jiaotong University. She is currently pursuing her MS degree in Computer Technology at East China Jiaotong University.

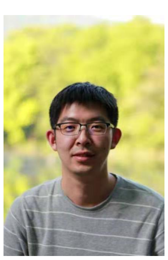
Her current research interests lie in Deep Learning, Natural Language Processing, and Social Networks.

**Jin Li** is currently a professor and vice dean of School of Computer Science, Guangzhou University. He received his B.S. (2002) and M.S. (2004) from Southwest University and Sun Yat-sen University, both in Mathematics. He got his Ph.D degree in information security from Sun Yat-sen University at 2007. He served as a senior research associate at Korea Advanced Institute of Technology (Korea) and Illinois Institute of Technology (U.S.A.) from 2008 to 2010, respectively. He has published more than 100 papers in international conferences and journals, including IEEE INFOCOM, IEEE TIFS, IEEE TPDS, IEEE TOC and ESORICS etc. His work has been cited more than 11000 times at Google Scholar and the H-Index is 40. He served as Associate editor for several international journals, including IEEE Transactions on Dependable and Secure Computing, Information Sciences. He also served as program chairs and committee for many international conferences such as CSS 2019, ICA3PP 2018, CSE 2017, IEEE EUC 2017, ISICA 2015 etc. He received several National Science Foundation of China (NSFC) Grants, including NSFC Outstanding Youth Foundation.

His research interests include Artificial Intelligence, Cloud Computing and Cryptographic Protocols.

**Xu Yuan** (S'13-M'16) received the B.S. degree from the Department of Information Security, Nankai University, in 2009, and the Ph.D. degree from the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA, in 2016. From 2016 to 2017, he was a Post-Doctoral Fellow of Electrical and Computer Engineering with the University of Toronto, Toronto, ON, Canada. He is currently an Assistant Professor in the School of Computing and Informatics at the University of Louisiana at Lafayette, LA, USA.

His research interest focuses on Cloud Computing, Social Network Security, and Cognitive Radio Networks.



**Jiaqi Zheng** is currently a Research Assistant Professor from Department of Computer Science and Technology, Nanjing University, China. He was a Research Assistant at the City University of Hong Kong in 2015, and collaborated with Huawei Noah's Ark Lab. He visited CIS center at Temple University in 2016. He received the Best Paper Award from IEEE ICNP 2015, Doctorial Dissertation Award from ACM SIGCOMM China 2018 and the First Prize of Jiangsu Science and Technology Award in 2018. He is a member of ACM and IEEE.

His research area is computer networking, particularly Data Center Networks, Machine Learning System and Online Optimization.