



# Discovering Protagonist of Sentiment with Aspect Reconstructed Capsule Network

Chi Xu<sup>1,3</sup>, Hao Feng<sup>4</sup>, Guoxin Yu<sup>1,2</sup>, Min Yang<sup>5</sup>, Xiting Wang<sup>6</sup>, Yan Song<sup>7</sup>,  
and Xiang Ao<sup>1,2</sup>✉

<sup>1</sup> Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China

<sup>2</sup> University of Chinese Academy of Sciences, Beijing 100049, China

<sup>3</sup> Beijing Institute of Computer Technology and Application, Beijing, China  
{yuguoxin20g,aoxiang}@ict.ac.cn

<sup>4</sup> University of Electronic Science and Technology of China, Chengdu, China

<sup>5</sup> Shenzhen Key Laboratory for High Performance Data Mining, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Beijing, China

<sup>6</sup> Microsoft Research Asia, Beijing, China

xitwan@microsoft.com

<sup>7</sup> The Chinese University of Hong Kong (Shenzhen), Shenzhen, China

**Abstract.** Most existing aspect-term level sentiment analysis (ATSA) approaches combined neural networks with attention mechanisms built upon given aspect to generate refined sentence representation for better predictions. In these methods, aspect terms are always provided in both training and testing process which may degrade aspect-level analysis into sentence-level prediction. However, the annotated aspect term might be unavailable in real-world scenarios which may challenge the applicability of the existing methods. In this paper, we aim to improve ATSA by discovering the potential aspect terms of the predicted sentiment polarity when the aspect terms of a test sentence are unknown. We access this goal by proposing a capsule network based model named CAPSAR. In CAPSAR, sentiment categories are denoted by capsules and aspect term information is injected into sentiment capsules through a sentiment-aspect reconstruction procedure during the training. As a result, coherent patterns between aspects and sentimental expressions are encapsulated by these sentiment capsules. Experiments on three widely used benchmarks demonstrate these patterns have potential in exploring aspect terms from test sentence when only feeding the sentence to the model. Meanwhile, the proposed CAPSAR can clearly outperform SOTA methods in standard ATSA tasks.

**Keywords:** Aspect-term level sentiment analysis · Capsule network · Sentiment-aspect reconstruction

---

C. Xu and H. Feng—Equally contributed.

© Springer Nature Switzerland AG 2021

C. S. Jensen et al. (Eds.): DASFAA 2021, LNCS 12682, pp. 120–135, 2021.

[https://doi.org/10.1007/978-3-030-73197-7\\_8](https://doi.org/10.1007/978-3-030-73197-7_8)

# 1 Introduction

Aspect-level sentiment analysis is an essential building block of sentiment analysis [24]. It aims at extracting and summarizing the sentiment polarities of given aspects of entities, i.e. *targets*, from customers’ comments. Two subtasks are explored in this field, namely Aspect-Term level Sentiment Analysis (ATSA) and Aspect-Category level Sentiment Analysis (ACSA). The purpose of ATSA is to predict the sentiment polarity with respect to given targets appearing in the text. For example, consider the sentence “*The camera of iPhone XI is delicate, but it is extremely expensive.*”, ATSA may ask the sentiment polarity towards the given target “*camera*”. Meanwhile, ACSA attempts to predict the sentiment tendency regarding a given target chosen from predefined categories, which may not explicitly appear in the comments. Take the same sentence as an example, ACSA asks the sentiment towards the aspect “*Price*” and derives a negative answer. In this paper, we aim at addressing the ATSA task.

Conventional approaches [5, 15, 19] incorporated linguistic knowledge, such as sentiment-lexicon, syntactic parser, and negation words, etc., and tedious feature engineering into the models to facilitate the prediction accuracy. Recently, supervised deep neural networks, e.g. Recurrent Neural Network (RNN) [36, 49], Convolution Neural Network (CNN) [21, 46] and attention mechanism [1, 8, 16, 28–30, 37, 39, 43] have shown remarkable successes without cumbersome feature designing. These models are able to effectively screen unrelated text spans and detect the sentiment context about the given target.

Despite these efforts, there is still a major deficiency in previous deep neural network based studies. Specifically, fully labeled aspect terms and their locations in sentence are explicitly required in both training and test process for recent methods, which may derive them degrade to sentence-level prediction and would fail for the test data without such annotations. To acquire aspect terms on predicted sentences, automatic aspect term detection may lead to error accumulation [45], and manually identifying is inefficient even infeasible. To support more authentic applications, it calls for an approach that is able to predict potential aspect-related sentiments based on the sentence and what it has learned from the training set. To this end, we propose a capsule network-based approach to remedy the above problem. Compared with previous studies, our method is able to explore featured sentiments so as to answer the question: “*What are the protagonists of the predicted sentiment polarity?*” We access this goal by leveraging the capsule network<sup>1</sup> [34], which has achieved promising results in computer vision [12, 34], natural language processing [7, 17, 50–52] and recommendation tasks [23].

The core idea of capsule network is the unit named *capsule*, which consists of a group of neurons in which its activity vector can represent the instantiation parameters of a specific type of entity. The length of activity vector denotes the

---

<sup>1</sup> Here we refer to the capsule network proposed by [34]. Though the models in [44] and [45] also called capsule network in their papers, they are basically built upon RNN and attention mechanisms with distinct concepts and implementations.

probability that the entity exists and its orientation can encode the properties of the entity. Inspired by that, we propose CAPSAR (**CAP**sule network with **Sentiment-Aspect Reconstruction**) framework by leveraging capsules to denote sentiment categories and enforce the potential aspect information as the corresponding properties. Specifically, during the training process, the sentence is first encoded with given aspects through a location proximity distillation. Then the encoded sentence representations are fed to hierarchical capsule layers and the final capsule layer represents all the concerned sentiment categories. To capture coherent patterns among aspects and sentimental expressions, the sentiment capsules are encouraged to encode the information about the aspect terms. We implement such procedure by reconstructing the aspect with the sentiment capsules. The reconstruction loss is taken as an additional regularization during the training. During the test phase, if the annotated aspect term is unseen by the model, CAPSAR can also make prediction and the potential aspect terms in the sentence could be detected by de-capsulizing the sentiment capsules. We evaluate the proposed methods on three widely used benchmarks. The results show the model has potential in unearthing aspect terms for new sentences, and it can also surpass SOTA baselines in standard aspect-term level sentiment analysis tasks.

## 2 Related Work

The related researches in literatures can be categorized as follows, including sentiment analysis based on neural network, aspect level sentiment classification and jointing learning methods for aspect level tasks.

**Sentiment Analysis Based on Neural Network.** Neural network approaches have achieved promising results on both document level [27, 41, 47] and sentence level [35] sentiment classification tasks without expensive feature engineering. Some works [10, 11] even exploited available interactions between document level and sentence level sentiment classification. [44] firstly adopted capsules into document-level sentiment analysis, but their capsule is still based on RNN and attentions, which is different from the capsule designs in [34].

**Aspect Level Sentiment Classification.** Aspect level sentiment classification is an emerging essential research topic in the field of sentiment analysis. The purpose is to infer the polarity with respect to aspect phrase or predefined aspect categories within the text. [36, 49] used multiple RNN layers to jointly model the relations between target terms and their left and right context. Attention-based methods were brought to this field by many researches [1, 3, 8, 9, 16, 20, 25, 29, 37, 38, 43] to exploit contextual and positional proximity of aspect terms for prediction and have achieved promising results. In addition, graph convolution networks (GCN) [48] were also utilized in this task. However, the final representation may still fail to capture the accurate

sentiment due to target-sensitive problem [42] or because of the noise in data. Models based on convolution neural networks [21, 46] are alternatives achieving competitive results, but some key information for modeling local meaning and overall sentiment may be blurred during the pooling operations. Recently, [45] proposed to use capsules to perform aspect-category level sentiment analysis. However as their previous work [44], the basic capsule module is based on attention mechanisms, which is entirely different with ours. And other works leveraging capsule network [7, 17] for aspect-level sentiment analysis require explicit aspect annotation during prediction while our method does not require.

**Joint Learning for Aspect Level Tasks.** There are some recent joint learning methods striving to combine different aspect level tasks into a unified learning process. For example, some studies proposed to extract aspect terms and predict corresponding sentiment polarities in a pipeline or an integrated model. The pipeline models [14, 33] are extract-then-classify processes and were proposed to solve the two tasks successively. For integrated models, [22, 40] extracted aspect terms with polarities by collapsed tagging that is a unified tagging scheme to link two tasks. [26] considered the relationship between the two tasks and attempted to investigate useful information from one task to another. In addition, some emerging methods [2, 11, 31] proposed to extract opinion words in sentences as auxiliary information to further improve the performance of aspect level sentiment classification.

### 3 The CAPSAR Model

#### 3.1 Model Overview

The overall architecture of CAPSAR is shown in Fig. 1. It starts from an embedding representation of words. In particular, we represent the  $i$ -th sentence in a dataset  $\mathbb{D}$  with  $m$  sentences as  $\{w_1^{(i)}, w_2^{(i)}, \dots, w_{n_i}^{(i)}\}$ , where  $i \in [1, \dots, m]$ ,  $n_i$  is the sentence length, and  $w^{(i)} \in W$  denotes a word where  $W$  is the set of vocabulary. The embedding layer encodes each word  $w_t^{(i)}$  into a real-value word vector  $x_t^{(i)} \in \mathbb{R}^{D_x}$  from a matrix  $M \in \mathbb{R}^{|W| \times D_x}$ , where  $|W|$  is the vocabulary size and  $D_x$  is the dimension of word vectors. The sentence is encoded by a sequence encoder to construct a sentence representation. Next, the output of the sequence encoder is fed to 3-layer capsules. The up-most capsule layer contains  $C$  *sentiment capsules*, where  $C$  is the number of sentiment categories. The capsules layers are communicated with a simple yet effective *sharing-weight* routing algorithm.

During training, one objective of our model is to maximize the length of sentiment capsules corresponding to the ground truth since it indicates the likelihood of potential sentiments. Meanwhile, these active vectors of the sentiment capsules are used to model the connections between the considered aspect terms and its corresponding sentiment via an aspect reconstruction. The distance between the reconstructed aspect representation produced by sentiment capsules and the

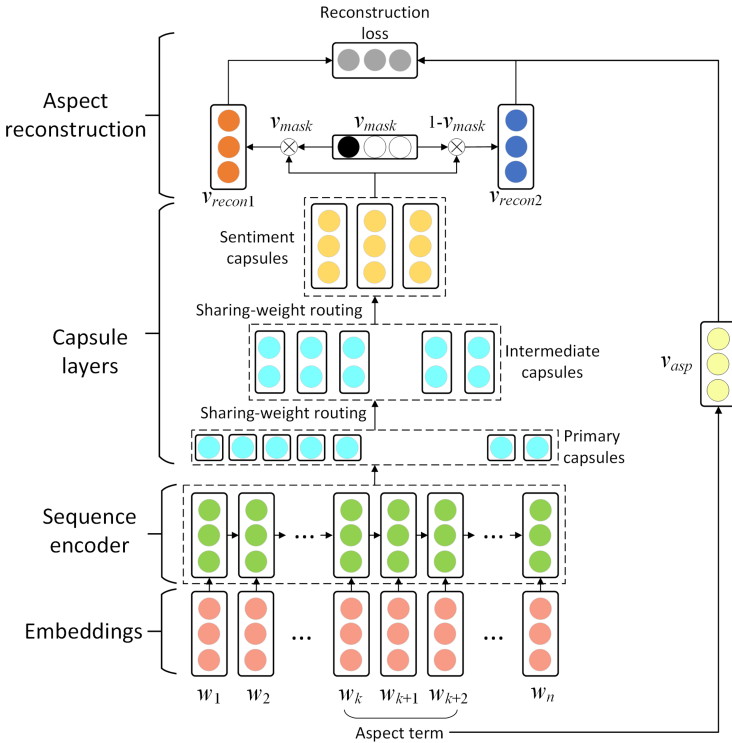


Fig. 1. The network architecture of CAPSAR.

given aspect embedding<sup>2</sup> is regarded as an additional regularization. In this manner, we encourage the sentiment capsules to learn the aspect information as their active vectors’ orientations. In the test process, a sentiment capsule will be “active” if its length is above a user-specific threshold, e.g. 0.5. All others will then be “inactive”. The sentiment prediction of a test sentence will be determined by the sentiment category associated with the active sentiment capsules.

### 3.2 Sequence Encoder

In our model, we adopt Bi-GRU as the sequence encoder for simplicity. For  $i$ -th sentence at step  $t$ , the corresponding hidden state  $h_t^{(i)}$  are updated as follows.

$$h_t^{(i)} = \begin{bmatrix} \vec{h}_t^{(i)} \\ \overleftarrow{h}_t^{(i)} \end{bmatrix} = \begin{bmatrix} \overrightarrow{\text{GRU}}(x_t^{(i)}) \\ \overleftarrow{\text{GRU}}(x_t^{(i)}) \end{bmatrix}, \quad t = 1, \dots, n_i, \quad (1)$$

where  $h_t^{(i)}$  concatenates hidden states of the  $t$ -th word in the  $i$ -th sentence from both directions. Note that more advanced encoders such as LSTM [13]

<sup>2</sup> The aspect embedding is calculated by the average of the word embeddings that form the aspect term.

or BERT [4], can also be utilized as the sequence encoder. We will introduce how to combine CAPSAR with BERT in the following part.

### 3.3 Location Proximity with Given Aspect

In order to highlight potential opinion words that are closer to given aspect terms, we adopt a location proximity strategy, which is observed effective in [1, 21]. Specifically, we calculate relevance  $l_t^{(i)}$  between the  $t$ -word and the aspect<sup>3</sup>.

$$l_t^{(i)} = \begin{cases} 1 + \max(0, \alpha + n_i/\beta - |\gamma * (k - t)|) & t \leq n_i \\ 0 & t > n_i \end{cases} \quad (2)$$

where  $k$  is the index of the first aspect word,  $n_i$  is the sentence length,  $\alpha$ ,  $\beta$  and  $\gamma$  are pre-specified constants.

We use  $l$  to help the sequence encoder locate possible key words w.r.t the given aspect.

$$\hat{h}_t^{(i)} = h_t^{(i)} * l_t^{(i)}, t \in [1, n_i], i \in [1, m] \quad (3)$$

Based on Eq. 2 and 3, the salience of words that are distant to the aspect terms will be declined. Note that such location proximity could be optional in the test process when the annotated aspect terms are unavailable.

### 3.4 Capsule Layers with Sharing-Weight Routing

The capsule layers of CAPSAR consist of a primary capsule layer, an intermediate capsule layer and a sentiment capsule layer. The primary capsule layer contains a group of neurons which are constructed by the hidden vectors of the sequence encoder. Specifically, we simply perform convolutional operation over  $h_{n_i}^{(i)}$  for  $i$ -th sentence and take its output to formulate the primary capsules. As a result, the primary capsules may contain the sentence coupled with aspect representations.

Next, the primary capsules are transformed into the intermediate layer and the subsequent sentiment capsule layer via a *sharing-weight* routing mechanism. Unlike the conventional dynamic routing algorithm in [34], our routing algorithm simultaneously keeps local-proximity information and significantly reduces training parameters.

The sharing-weight routing algorithm shares the weights between different children and the same parent. Specifically, we denote the output vector of a capsule  $i$  at level  $L$  and the total input vector of a capsule  $j$  at level  $L + 1$  as  $p_i \in \mathbb{R}^{D_L}$  and  $\tilde{q}_j \in \mathbb{R}^{D_{L+1}}$ , respectively. We use a unified transformation weight matrix  $W_j \in \mathbb{R}^{D_{L+1} \times D_L}$  for the capsule  $j$  at level  $L+1$  to compute the prediction vectors, i.e.  $\hat{p}_{j|i} \in \mathbb{R}^{D_{L+1}}$ , for every possible child capsule  $i$  at level  $L$ . As a result, the total input  $\tilde{q}_j$  of capsule  $j$  is updated as

$$\tilde{q}_j = \sum_i c_{ij} \hat{p}_{j|i}, \hat{p}_{j|i} = W_j p_i \quad (4)$$

<sup>3</sup>  $t$  is possibly larger than  $n_i$  because of sentence padding.

where  $c_{ij}$  denotes coupling coefficients between capsule  $i$  and  $j$  and is initialized with equal probability. During the iterative dynamic routing process,  $c_{ij}$  is updated to  $q_j \cdot \hat{p}_{j|i}$ , where  $q_j$  is the output vector of capsule  $j$  computed by the squash function.

$$q_j = \frac{\|\tilde{q}_j\|^2}{1 + \|\tilde{q}_j\|^2} \frac{\tilde{q}_j}{\|\tilde{q}_j\|} \quad (5)$$

Compared with the conventional dynamic routing algorithm, the sharing-weight routing algorithm clearly reduces the number of parameters and saves computational cost. For example, if two consecutive capsule layers have  $M$  and  $N$  capsules and the dimensions are  $D_L$  and  $D_{L+1}$  respectively, then the number of parameters to be learned in this layer will be reduced by  $(M-1) \times N \times D_L \times D_{L+1}$  compared with the original routing algorithm in [34].

### 3.5 Model Training with Aspect Reconstruction

The training objective of this model is two-fold. On one hand, we aim to maximize the length of the correct sentiment capsules since it indicates the probability that the corresponding sentiment exists. To this end, we use a margin loss for every given sentence  $i$

$$L_1^{(i)} = v_{mask}^{(i)} \max(0, m^+ - \|v_{prob}^{(i)}\|)^2 + (1 - v_{mask}^{(i)}) \max(0, \|v_{prob}^{(i)}\| - m^-)^2 \quad (6)$$

Here  $v_{prob}^{(i)} = (\|q_1^{(i)}\|, \dots, \|q_C^{(i)}\|)$  where  $q_j^{(i)}$  denotes the output vector of sentiment capsule  $j$  for the sentence  $i$ . Each element in such  $v_{prob}^{(i)}$  indicates the existence probability of the corresponding sentiment in sentence  $i$ ;  $v_{mask}^{(i)}$  is the mask for sentence  $i$ ;  $m^+$  and  $m^-$  are hyper-parameters.

On the other hand, we attempt to encourage the sentiment capsules to capture interactive patterns between the aspect and their corresponding sentiments. To this end, we utilize the output vectors of all the sentiment capsules  $q_j$  ( $j \in [1, C]$ ) to participate in reconstructing the representation of aspect terms. Specifically, suppose  $v_{mask}$  is a one-hot mask<sup>4</sup> whose element representing the ground truth sentiment is 1, and the rest values are 0. Then we derive two vectors, namely  $v_{recon1}$  and  $v_{recon2}$ , through this mask and the sentiment capsules. First, we mask out all but the output vector of the correct sentiment capsule by  $v_{mask}$ . Then  $v_{recon1}$  is derived by transforming the masked output vector through a fully-connect layer.  $v_{recon2}$  can be derived in a similar manner where  $1 - v_{mask}$  is used as the mask. We force both  $v_{recon1}$  and  $v_{recon2}$  have the same dimension of word embedding, i.e.,  $v_{recon1}, v_{recon2} \in \mathbb{R}^{D_x}$ , and they are contributed to the aspect reconstruction during the training.

Suppose a given aspect embedding<sup>5</sup> of the sentence  $i$  is denoted as  $v_{asp}^{(i)}$ , another training objective in our CAPSAR is to minimize the distance between

<sup>4</sup> The dimension of  $v_{mask}$  is  $C$ .

<sup>5</sup> If there are more than one aspect in a same sentence, every aspect will be separately trained.

$v_{asp}$  and  $v_{recon1}$ , and to maximize that between  $v_{asp}$  and  $v_{recon2}$ .

$$L_2^{(i)} = -v_{asp}^{(i)} \frac{v_{recon1}^{(i)}}{\|v_{recon1}^{(i)}\|} + v_{asp}^{(i)} \frac{v_{recon2}^{(i)}}{\|v_{recon2}^{(i)}\|} \quad (7)$$

Finally, the overall loss is the combination of  $L_1^{(i)}$  and  $L_2^{(i)}$ , and a hyper-parameter  $\lambda$  is used to adjust the weight of  $L_2$ .

$$Loss = \sum_i (L_1^{(i)} + \lambda L_2^{(i)}) \quad (8)$$

For prediction, a sentence and an optional aspect in the sentence are fed to the network and the polarity attached to the sentiment capsule with the largest length will be assigned.

### 3.6 Combining CAPSAR with BERT

Our CAPSAR is meanwhile easily extended that utilizes the features learnt from large-scale pre-trained encoders, e.g. BERT [4]. An upgraded model, namely CAPSAR-BERT, is achieved by replacing the sequence encoder with BERT in CAPSAR. The other structures are kept the same. In this manner, the strength of BERT and the proposed structures could be combined.

## 4 Experiments

In this section, we verify the effectiveness of CAPSAR. Firstly, we verify the ability of CAPSAR on perceiving the potential aspect terms when they are unknown. Secondly, we investigate the performance of CAPSAR on standard ATSA tasks, where the aspect terms are always known for test sentences. Finally, we demonstrate the detailed differences of compared methods by case studies.

### 4.1 Datasets

Three widely used benchmark datasets are adopted in the experiment whose statistics are shown in Table 1. **Restaurant** and **Laptop** are from SemEval2014 Task 4<sup>6</sup>, which contain reviews from Restaurant and Laptop domains, respectively. We delete a tiny amount of data with conflict labels which follows previous works [37, 43]. **Twitter** is collected by [6] containing twitter posts. Though these three benchmarks are not large-scale datasets, they are the most popular and fair test beds for recent methods.

<sup>6</sup> <http://alt.qcri.org/semeval2014/task4/index.php?id=data-and-tools>.



**Table 1.** Statistics of datasets.

Dataset	Restaurant		Laptop		Twitter	
	Train	Test	Train	Test	Train	Test
Neg.	807	196	870	128	1562	173
Neu.	637	196	464	169	3124	346
Pos.	2164	728	994	341	1562	173

## 4.2 Compared Methods

We compare our method with several SOTA approaches.

**ATAE-LSTM** [43] appends aspect embedding with each input word embeddings. **TD-LSTM** [36] employs two LSTMs to model contexts of the targets and performs predictions based on the concatenated context representations. **IAN** [29] interactively learns the context and target representation. **Mem-Net** [37] feeds target word embedding to multi-hop memory to learn better representations. **RAM** [1] uses recurrent attention to capture key information on a customized memory. **MGAN** [8] equips a multi-grained attention to address the aspect having multiple words or larger context. **ANTM** [30] adopts attentive neural Turing machines to learn dependable correlation of aspects to context. **CAPSAR** and **CAPSAR-BERT** are models proposed in this paper. **BERT** [4] is also compared to show the improvement of CAPSAR-BERT.

## 4.3 Experimental Settings

In our experiments, we implement our method by Keras 2.2.4. The word embedding is initialized by Glove 42B [32] with dimension of 300. The max length for each sentence is set to 75. The batch size of training is 64 for 80 epochs, and Adam [18] with default setting is taken as the optimizer. For sequence encoder, we adopt the Bi-GRU with the dropout rate of 0.5 for CAPSAR. The pre-trained BERT of the dimension of 768 is used in CAPSAR-BERT. For hyper-parameters in Eq. 2,  $\alpha$ ,  $\beta$ , and  $\gamma$  are set to be 3, 10 and 1. For the capsule layers, there are 450 primary capsules with dimensions of 50, 30 intermediate capsules with dimensions of 150 and 3 sentiment capsules with dimensions of 300. The default routing number is 3. For hyper-parameters in the loss function (cf. Eq. 8), we set  $m^+$ ,  $m^-$ ,  $\lambda$  to be 1.0, 0.1 and 0.003 respectively. Evaluation metrics we adopt are Accuracy and Macro-F1, and the latter is widely used for recent ATSA tasks since it is more appropriate for datasets with unbalanced classes.

## 4.4 Results on Standard ATSA

**Main Results.** Table 2 demonstrates the performances of compared methods over the three datasets on the standard ASTA tasks. On such setting, every aspect term is known to all the models, and each model predicts the corresponding polarity for a given aspect term. Here we only consider the longest sentiment

**Table 2.** The average accuracy and macro F1-score on standard ATSA tasks. The results with ‘\*’ are retrieved from the papers of RAM, and other results of baselines are retrieved from corresponding papers.

	Models	Restaurant		Laptop		Twitter	
		Accuracy	F1	Accuracy	F1	Accuracy	F1
Baselines	ATAE-LSTM	0.7720	NA	0.6870	NA	NA	NA
	TD-LSTM	0.7560	NA	0.6810	NA	0.6662*	0.6401*
	IAN	0.7860	NA	0.7210	NA	NA	NA
	MemNet(3)	0.8032	NA	0.7237	NA	0.6850*	0.6691*
	RAM(3)	0.8023	0.7080	0.7449	0.7135	0.6936	0.6730
	MGAN	0.8125	0.7194	0.7539	<b>0.7247</b>	0.7254	0.7081
	ANTM	0.8143	0.7120	0.7491	0.7142	0.7011	0.6814
Ablation test	CAPSAR w/o R	0.8185	0.7216	0.7484	0.7039	0.7255	0.7067
	CAPSAR w/o H	0.8188	0.7226	0.7461	0.7054	0.7298	0.7080
	CAPSAR	<b>0.8286</b>	<b>0.7432</b>	<b>0.7593</b>	0.7221	<b>0.7368</b>	<b>0.7231</b>
Combine BERT	BERT	0.8476	0.7713	0.7787	0.7371	0.7537	0.7383
	CAPSAR-BERT	<b>0.8594</b>	<b>0.7867</b>	<b>0.7874</b>	<b>0.7479</b>	<b>0.7630</b>	<b>0.7511</b>

capsule is active. All the reported values of our methods are the average of 5 runs to eliminate the fluctuates with different random initialization, and the performance of baselines are retrieved from their papers for fair comparisons. The best performances are demonstrated in bold face. From the table, we observe CAPSAR has clear advantages over baselines on all datasets. Our model can outperform all the baselines by a large-margin on both evaluate measures except the F1 on Laptop dataset. Meanwhile, we also observe that CAPSAR-BERT further improves the performance of BERT. It demonstrates the advantages by combining CAPSAR with advanced pre-trained model.

**Ablation Test.** Next we perform ablation test to show the effectiveness of component of CAPSAR. We remove the aspect reconstruction and intermediate capsule layer, respectively, and derive two degrade models, namely **CAPSAR w/o R** and **CAPSAR w/o H**. Their performance on the three datasets are exhibited in Table 2, and we can observe the degraded model gives a clear weaker performance than CAPSAR, without the sentiment-aspect reconstruction. We conjecture such a regularizer might be able to learn the interactive patterns among the aspects with the complex sentimental expressions. Meanwhile, we also observe the advantage of the intermediate capsule layer, which illustrates the hierarchical capsule layers may be stronger in learning aspect-level sentiment features.

**Case Studies.** Then we illustrate several case studies on the results of ATSA task in Table 3. The predictions of CAPSAR, ANTM, MGAN and RAM are exhibited. We directly run their available models on test set to get the results. The input aspect terms are placed in the brackets with their true polarity labels

**Table 3.** Prediction examples of some of the compared methods. The abbreviations Pos, Neu and Neg in the table represent positive, neutral and negative.  $\times$  indicates incorrect prediction.

Sentence	CAPSAR	ANTM	MGAN	RAM
1. The [ <b>chocolate raspberry cake</b> ] <sub>Pos</sub> is heavenly - not too sweet , but full of [ <b>flavor</b> ] <sub>Pos</sub> .	(Pos,Pos)	(Pos,Neg $\times$ )	(Neg $\times$ ,Pos)	(Pos,Neg $\times$ )
2. Not only was the sushi fresh , they also served other [ <b>entrees</b> ] <sub>Neu</sub> allowed each guest something to choose from and we all left happy (try the [ <b>duck</b> ] <sub>Pos</sub> !	(Neu,Pos)	(Pos $\times$ ,Pos)	(Pos $\times$ ,Pos)	(Pos $\times$ ,Pos)
3. The [ <b>baterry</b> ] <sub>Pos</sub> is very longer .	Pos	Neg $\times$	Pos	Neg $\times$
4. [ <b>Startup times</b> ] <sub>Neg</sub> are incredibly long : over two minutes.	Neg	Neg	Pos $\times$	Pos $\times$
5. However I chose two day [ <b>shipping</b> ] <sub>Neg</sub> and it took over a week to arrive.	Neu $\times$	Neu $\times$	Pos $\times$	Pos $\times$

as subscripts. “Pos”, “Neu” and “Neg” in the table represent positive, neutral and negative, respectively. Different targets are demonstrated by different colors, such as blue and red, etc. The context which may support the sentiment of targets is manually annotated and dyed with the corresponding color.

For instance, in the first sentence, one target is “*chocolate raspberry cake*” of which the sentiment is positive and the context “*is heavenly*” supports this sentiment. In this case, the results of CAPSAR, ANTM, and RAM are correct. For the second aspect “*flavor*”, our method and MGAN predict correctly. We conjecture the reason is the sentence contains turns in its expression, which may confuse the existing neural network models. Similar observation is achieved on the second sentence in which only our method can predict different sentiments for distinct aspects.

Next we discuss a target-sensitive case which is shown by the third and the fourth sentences in the same table. The word “long” in the exhibited two sentences indicates entirely opposite sentiment polarities, since the expressed sentiment also depends on the considered aspects. These cases are challenge for algorithms to identify the sentiment of each sentence correctly. Among the demonstrated methods, only our approach can predict them all successfully. ANTM, as a strong competitor, can predict one of them correctly. The others fail to give any correct prediction. We do not claim our CAPSAR can perfectly address the target-sensitive cases, however, the results give an initial encouraging potential. We argue that it is the sentiment-aspect reconstruction in our model which makes aspect and its corresponding sentiment become more coupled, and it might be one of the essential reasons why our model achieves a

**Table 4.** The average Precision@1, Recall@5 and mAP on aspect term detection. The column “Avg. Aspect” and “Avg.SenLen” indicate average number of words on aspect terms in each sentence and average length of each sentence on the test set, respectively.

Datasets	Avg. Aspect	Avg. SenLen	Pre.@1	Rec.@5	mAP
Restaurant	2.76	16.25	0.8233	0.7884	0.7139
Laptop	2.54	15.79	0.6408	0.7557	0.6173

salient improvement on ATSA task. How to specifically explore capsule networks for target-sensitive sentiment analysis is out of the scope of this paper.

For error analysis, we find that all the listed models cannot predict correctly on the last sentence of Table 3. By looking closer to this sentence, we recognize that the sentiment polarity of this sentence comes from implicit semantics instead of its explicit opinion words. It indicates implicit semantics inference behind sentences is still a major challenge of neural network models, even exploiting capsule networks.

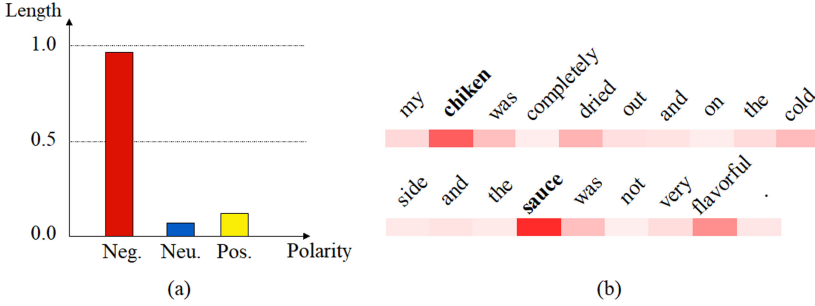
#### 4.5 Results on Aspect Term Detection

Next, we investigate whether CAPSAR can detect potential aspect terms when they are unknown during the test. We use the trained model to predict every test sentence on Restaurant and Laptop datasets but we intentionally conceal the information about the aspect terms in the input. In another word, the model is only fed the test sentence without any other additional input. Then we de-capsulize the sentiment capsule whose length is longer than 0.5 and compute normalized dot-product between its reconstructed vector and every word embedding in the test sentence. These dot-products can be regarded as the probabilities representing the possibility of the word to be part of an aspect term.

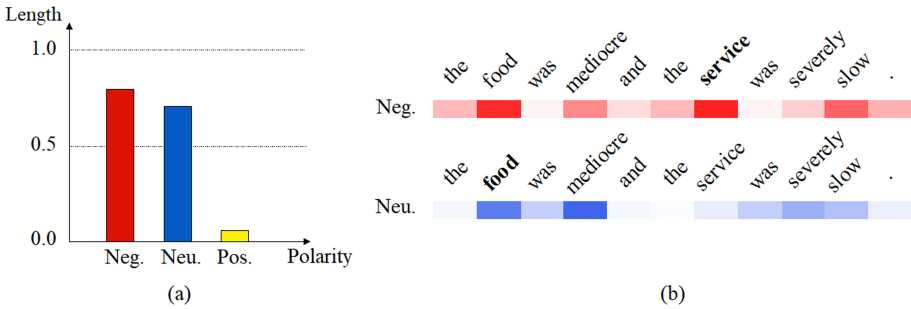
A sentence may simultaneously contain multiple sentiments (c.f. the 2nd sentence in Table 3), which derives more than one sentiment capsule whose length surpasses 0.5. As a result, we detect the potential aspect terms for every active sentiment capsule, respectively, in our evaluation. There could be more than one aspect terms for each sentiment category in the same sentence (c.f. the 1st sentence in Table 3). Hence we compute Precision@ $k$ , Recall@ $k$  and mean Average Precision (mAP) to comprehensively verify the effectiveness of CAPSAR on aspect term detection.

We retrain our CAPSAR five times and use the trained model to detect aspect terms on the test set. Table 4 shows the average results. From the table we observe that CAPSAR shows an encouraging ability to extract aspect terms even though they are unknown in new sentences. Meanwhile, the model achieves better performance on Restaurant dataset. We conjecture the reason is the Laptop dataset has more complicated aspect terms such as “Windows 7”.

Finally, we visualize two test sentences in this evaluation shown as Fig. 2 and 3. Figure 2 demonstrates a case that only one sentiment capsule is active.



**Fig. 2.** The visualization of aspect term detection when single sentiment capsule is active. The real aspect terms are marked in bold face in the sub-figure (b).



**Fig. 3.** The visualization of aspect term detection when multiple sentiment capsules are active. The real aspect terms are marked in bold face in the sub-figure (b), respectively.

The sentiment capsule length is exhibited in Fig. 2(a), and the corresponding dot-products are demonstrated as Fig. 2(b). The darker the color in Fig. 2(b), the higher value the dot-product is, which means the corresponding word is more likely to be a part of an aspect term. From this figure, we observe that the two real aspect terms hold much higher weights compared with the other words, which derives a correct detection. We can also obtain similar observations from the case shown as Fig. 3 in which two sentiment capsules are active. In this case, there are two aspect terms, namely “food” and “service”. From the sentence, we observe that this review has a strong negative sentiment towards on “service” while has a neutral attitude to “food”. After we de-capsulize the active sentiment capsules, we find both “service” and “food” are perceived by the negative capsule because of the overall negative sentiment of the whole sentence. While in the neutral capsule, only the corresponding aspect term “food” is highlighted.

## 5 Conclusion

In this paper, we proposed, CAPSAR, a capsule network based model for improving aspect-level sentiment analysis. The network is piled up hierarchical capsule

layers equipped with a shared-weight routing algorithm to capture key features for predicting sentiment polarities. Meanwhile, the instantiation parameters of sentiment capsules are used to reconstruct the aspect representation, and the reconstruction loss is taken as a part of the training objective. As a consequence, CAPSAR could further capture the coherent patterns between sentiment and aspect information and is able to detect potential aspect terms by parsing the sentiment capsules when these aspect terms are unseen. Experimental results on three real-world benchmarks demonstrate the superiority of the proposed model.

**Acknowledgments.** This work is supported by the National Natural Science Foundation of China under Grant No. 61976204, 92046003 and U1811461, the Project of Youth Innovation Promotion Association CAS and Beijing Nova Program Z201100006820062. This work was also supported by the Natural Science Foundation of Chongqing under Grant No. cstc2019jcyj-msxmX0149. Min Yang is partially supported by Shenzhen Basic Research Foundation (No. JCYJ20200109113441941).

## References

1. Chen, P., Sun, Z., Bing, L., Yang, W.: Recurrent attention network on memory for aspect sentiment analysis. In: EMNLP (2017)
2. Chen, Z., Qian, T.: Relation-aware collaborative learning for unified aspect-based sentiment analysis. In: ACL, pp. 3685–3694 (2020)
3. Cheng, J., Zhao, S., Zhang, J., King, I., Zhang, X., Wang, H.: Aspect-level sentiment classification with heat (hierarchical attention) network. In: CIKM, pp. 97–106 (2017)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL (2019)
5. Ding, X., Liu, B., Yu, P.S.: A holistic lexicon-based approach to opinion mining. In: WSDM (2008)
6. Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., Xu, K.: Adaptive recursive neural network for target-dependent Twitter sentiment classification. In: ACL (2014)
7. Du, C., et al.: Capsule network with interactive attention for aspect-level sentiment classification. In: EMNLP-IJCNLP (2019)
8. Fan, F., Feng, Y., Zhao, D.: Multi-grained attention network for aspect-level sentiment classification. In: EMNLP (2018)
9. He, R., Lee, W.S., Ng, H.T., Dahlmeier, D.: Effective attention modeling for aspect-level sentiment classification. In: COLING, pp. 1121–1131 (2018)
10. He, R., Lee, W.S., Ng, H.T., Dahlmeier, D.: Exploiting document knowledge for aspect-level sentiment classification. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Melbourne, Australia, pp. 579–585. ACL (July 2018)
11. He, R., Lee, W.S., Ng, H.T., Dahlmeier, D.: An interactive multi-task learning network for end-to-end aspect-based sentiment analysis. In: ACL, pp. 504–515 (2019)
12. Hinton, G.E., Sabour, S., Frosst, N.: Matrix capsules with em routing. In: ICLR (2018)
13. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**, 1735–1780 (1997)

14. Hu, M., Peng, Y., Huang, Z., Li, D., Lv, Y.: Open-domain targeted sentiment analysis via span-based extraction and classification. In: ACL, pp. 537–546 (2019)
15. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: KDD (2004)
16. Huang, B., Ou, Y., Carley, K.M.: Aspect level sentiment classification with attention-over-attention neural networks. In: SBP-BRiMS (2018)
17. Jiang, Q., Chen, L., Xu, R., Ao, X., Yang, M.: A challenge dataset and effective models for aspect-based sentiment analysis. In: EMNLP (2019)
18. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: ICLR (2015)
19. Kiritchenko, S., Zhu, X., Cherry, C., Mohammad, S.: NRC-Canada-2014: detecting aspects and sentiment in customer reviews. In: Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval 2014 (2014)
20. Lei, Z., Yang, Y., Yang, M., Zhao, W., Guo, J., Liu, Y.: A human-like semantic cognition network for aspect-level sentiment classification. *AAAI* **33**, 6650–6657 (2019)
21. Li, X., Bing, L., Lam, W., Shi, B.: Transformation networks for target-oriented sentiment classification. In: ACL (2018)
22. Li, X., Bing, L., Li, P., Lam, W.: A unified model for opinion target extraction and target sentiment prediction. *AAAI* **33**, 6714–6721 (2019)
23. Li, C., Quan, C., Li, P., Qi, Y., Deng, Y., Wu, L.: A capsule network for recommendation and explaining what you like and dislike. In: SIGIR (2019)
24. Liu, B.: Sentiment analysis and opinion mining. In: Synthesis Lectures on Human Language Technologies (2012)
25. Liu, J., Zhang, Y.: Attention modeling for targeted sentiment. In: ACL, pp. 572–577 (2017)
26. Luo, H., Li, T., Liu, B., Zhang, J.: DOER: dual cross-shared RNN for aspect term-polarity co-extraction. In: ACL, pp. 591–601 (2019)
27. Luo, L., et al.: Beyond polarity: interpretable financial sentiment analysis with hierarchical query-driven attention. In: IJCAI (2018)
28. Luo, L., et al.: Unsupervised neural aspect extraction with sememes. In: IJCAI (2019)
29. Ma, D., Li, S., Zhang, X., Wang, H.: Interactive attention networks for aspect-level sentiment classification. In: IJCAI (2017)
30. Mao, Q., et al.: Aspect-based sentiment classification with attentive neural turing machines. In: IJCAI (2019)
31. Peng, H., Xu, L., Bing, L., Huang, F., Lu, W., Si, L.: Knowing what, how and why: a near complete solution for aspect-based sentiment analysis. In: *AAAI*, pp. 8600–8607 (2020)
32. Pennington, J., Socher, R., Manning, C.: GloVe: global vectors for word representation. In: EMNLP (2014)
33. Phan, M.H., Ogunbona, P.O.: Modelling context and syntactical features for aspect-based sentiment analysis. In: ACL, pp. 3211–3220 (2020)
34. Sabour, S., Frosst, N., Hinton, G.E.: Dynamic routing between capsules. In: NIPS (2017)
35. Socher, R., Pennington, J., Huang, E.H., Ng, A.Y., Manning, C.D.: Semi-supervised recursive autoencoders for predicting sentiment distributions. In: EMNLP (2011)
36. Tang, D., Qin, B., Feng, X., Liu, T.: Effective LSTMs for target-dependent sentiment classification. In: COLING (2016)
37. Tang, D., Qin, B., Liu, T.: Aspect level sentiment classification with deep memory network. In: EMNLP (2016)

38. Tang, J., et al.: Progressive self-supervised attention learning for aspect-level sentiment analysis. In: ACL, pp. 557–566 (2019)
39. Wang, B., Lu, W.: Learning latent opinions for aspect-level sentiment classification. In: AAAI (2018)
40. Wang, F., Lan, M., Wang, W.: Towards a one-stop solution to both aspect extraction and sentiment analysis tasks with neural multi-task learning. In: IJCNN, pp. 1–8 (2018)
41. Wang, J., Wang, Z., Zhang, D., Yan, J.: Combining knowledge with deep convolutional neural networks for short text classification. In: IJCAI (2017)
42. Wang, S., Mazumder, S., Liu, B., Zhou, M., Chang, Y.: Target-sensitive memory networks for aspect sentiment classification. In: ACL (2018)
43. Wang, Y., Huang, M., Zhao, L., et al.: Attention-based LSTM for aspect-level sentiment classification. In: EMNLP (2016)
44. Wang, Y., Sun, A., Han, J., Liu, Y., Zhu, X.: Sentiment analysis by capsules. In: The Web Conference (2018)
45. Wang, Y., Sun, A., Huang, M., Zhu, X.: Aspect-level sentiment analysis using AS-capsules. In: The Web Conference (2019)
46. Xue, W., Li, T.: Aspect based sentiment analysis with gated convolutional networks. In: ACL (2018)
47. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A.J., Hovy, E.H.: Hierarchical attention networks for document classification. In: HLT-NAACL (2016)
48. Zhang, C., Li, Q., Song, D.: Aspect-based sentiment classification with aspect-specific graph convolutional networks. In: EMNLP, pp. 4560–4570 (2019)
49. Zhang, M., Zhang, Y., Vo, D.T.: Gated neural networks for targeted sentiment analysis. In: AAAI (2016)
50. Zhang, N., Deng, S., Sun, Z., Chen, X., Zhang, W., Chen, H.: Attention-based capsule networks with dynamic routing for relation extraction. In: EMNLP (2018)
51. Zhang, X., Li, P., Jia, W., Zhao, H.: Multi-labeled relation extraction with attentive capsule network. In: AAAI (2019)
52. Zhao, W., Ye, J., Yang, M., Lei, Z., Zhang, S., Zhao, Z.: Investigating capsule networks with dynamic routing for text classification. In: EMNLP (2018)