

Multi-Instance Multi-Label Learning Networks for Aspect-Category Sentiment Analysis

Yuncong Li^{1*}, Cunxiang Yin^{1*}, Sheng-hua Zhong^{2†}, Xu Pan¹

¹Baidu Inc., Beijing, China

{liyuncong, yincunxiang, panxu}@baidu.com

²College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China

csszhong@szu.edu.cn

Abstract

Aspect-category sentiment analysis (ACSA) aims to predict sentiment polarities of sentences with respect to given aspect categories. To detect the sentiment toward a particular aspect category in a sentence, most previous methods first generate an aspect category-specific sentence representation for the aspect category, then predict the sentiment polarity based on the representation. These methods ignore the fact that the sentiment of an aspect category mentioned in a sentence is an aggregation of the sentiments of the words indicating the aspect category in the sentence, which leads to suboptimal performance. In this paper, we propose a Multi-Instance Multi-Label Learning Network for Aspect-Category sentiment analysis (AC-MIMLLN), which treats sentences as bags, words as instances, and the words indicating an aspect category as the key instances of the aspect category. Given a sentence and the aspect categories mentioned in the sentence, AC-MIMLLN first predicts the sentiments of the instances, then finds the key instances for the aspect categories, finally obtains the sentiments of the sentence toward the aspect categories by aggregating the key instance sentiments. Experimental results on three public datasets demonstrate the effectiveness of AC-MIMLLN¹.

1 Introduction

Sentiment analysis (Pang and Lee, 2008; Liu, 2012) has attracted increasing attention recently. Aspect-based sentiment analysis (ABSA) (Pontiki et al., 2014, 2015, 2016) is a fine-grained sentiment analysis task and includes many subtasks, two of which are aspect category detection (ACD) that detects the aspect categories mentioned in a sentence and

*Equal contribution

†Corresponding author

¹Data and code are available at <https://github.com/l294265421/AC-MIMLLN>

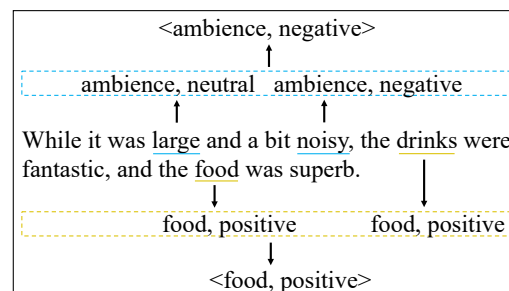


Figure 1: An example of ACD and ACSA. The underlined words are key instances, the labels of the key instances are in the dotted line boxes, and the labels of the sentence are in the angle brackets.

aspect-category sentiment analysis (ACSA) that predicts the sentiment polarities with respect to the detected aspect categories. Figure 1 shows an example. ACD detects the two aspect categories, *ambience* and *food*, and ACSA predicts the negative and positive sentiment toward them respectively. In this work, we focus on ACSA, while ACD as an auxiliary task is used to find the words indicating the aspect categories in sentences for ACSA.

Since a sentence usually contains one or more aspect categories, previous studies have developed various methods for generating aspect category-specific sentence representations to detect the sentiment toward a particular aspect category in a sentence. To name a few, attention-based models (Wang et al., 2016; Cheng et al., 2017; Tay et al., 2018; Hu et al., 2019) allocate the appropriate sentiment words for the given aspect category. Xue and Li (2018) proposed to generate aspect category-specific representations based on convolutional neural networks and gating mechanisms. Since aspect-related information may already be discarded and aspect-irrelevant information may be retained in an aspect independent encoder, some existing methods (Xing et al., 2019; Liang et al., 2019) utilized the given aspect to guide the sentence encoding from

scratch. Recently, BERT based models (Sun et al., 2019; Jiang et al., 2019) have obtained promising performance on the ACSA task. However, these models ignored that the sentiment of an aspect category mentioned in a sentence is an aggregation of the sentiments of the words indicating the aspect category. It leads to suboptimal performance of these models. For the example in Figure 1, both “drinks” and “food” indicate the aspect category *food*. The sentiment about *food* is a combination of the sentiments of “drinks” and “food”. Note that, words indicating aspect categories not only contain aspect terms explicitly indicating an aspect category but also contain other words implicitly indicating an aspect category (Cheng et al., 2017). In Figure 1, while “drinks” and “food” are aspect terms explicitly indicating the aspect category *food*, “large” and “noisy” are not aspect terms implicitly indicating the aspect category *ambience*.

In this paper, we propose a Multi-Instance Multi-label Learning Network for Aspect-Category sentiment analysis (AC-MIMLLN). AC-MIMLLN explicitly models the fact that the sentiment of an aspect category mentioned in a sentence is an aggregation of the sentiments of the words indicating the aspect category. Specifically, AC-MIMLLN treats sentences as bags, words as instances, and the words indicating an aspect category as the key instances (Liu et al., 2012) of the aspect category. Given a bag and the aspect categories mentioned in the bag, AC-MIMLLN first predicts the instance sentiments, then finds the key instances for the aspect categories, finally aggregates the sentiments of the key instances to get the bag-level sentiments of the aspect categories.

Our main contributions can be summarized as follows:

- We propose a Multi-Instance Multi-Label Learning Network for Aspect-Category sentiment analysis (AC-MIMLLN). AC-MIMLLN explicitly model the process that the sentiment of an aspect category mentioned in a sentence is obtained by aggregating the sentiments of the words indicating the aspect category.
- To the best of our knowledge, it is the first time to explore multi-instance multi-label learning in aspect-category sentiment analysis.
- Experimental results on three public datasets demonstrate the effectiveness of AC-MIMLLN.

2 Related Work

Aspect-Category Sentiment Analysis predicts the sentiment polarities with regard to the given aspect categories. Many methods have been developed for this task. Wang et al. (2016) proposed an attention-based LSTM network, which can concentrate on different parts of a sentence when different aspect categories are taken as input. Some new attention-based methods (Cheng et al., 2017; Tay et al., 2018; Hu et al., 2019) allocated more appropriate sentiment words for aspect categories and obtained better performance. Ruder et al. (2016) modeled the interdependencies of sentences in a text with a hierarchical bidirectional LSTM. Xue and Li (2018) extracted sentiment features with convolutional neural networks and selectively outputted aspect category related features with gating mechanisms. Xing et al. (2019), Liang et al. (2019) and Zhu et al. (2019) incorporated aspect category information into sentence encoders in the context modeling stage. Lei et al. (2019) proposed a human-like semantic cognition network to simulate the human beings’ reading cognitive process. Sun et al. (2019) constructed an auxiliary sentence from the aspect category and converted ACSA to a sentence-pair classification task. Jiang et al. (2019) put forward new capsule networks to model the complicated relationship between aspect categories and contexts. The capsule networks achieved state-of-the-art results. Several joint models (Li et al., 2017; Schmitt et al., 2018; Wang et al., 2019; Li et al., 2019) were proposed to avoid error propagation, which performed ACD and ACSA jointly.

However, all these models mentioned above ignored that the sentiment of an aspect category discussed in a sentence is an aggregation of the sentiments of the words indicating the aspect category.

Multi-Instance Multi-Label Learning (MIMLL) (Zhou and Zhang, 2006) deals with problems where a training example is described by multiple instances and associated with multiple class labels. MIMLL has achieved success in various applications due to its advantages on learning with complicated objects, such as image classification (Zhou and Zhang, 2006; Chen et al., 2013), text categorization (Zhang and Zhou, 2008), relation extraction (Surdeanu et al., 2012; Jiang et al., 2016), etc. In ACSA, a sentence contains multiple words (instances) and expresses sentiments to multiple aspect categories (labels), so MIMLL is suitable for ACSA. However, as far

as our knowledge, MIMLL has not been explored in ACSA.

Multiple instance learning (MIL) (Keeler and Rumelhart, 1992) is a special case of MIMLL, where a real-world object described by a number of instances is associated with only one class label. Some studies (Kotzias et al., 2015; Angelidis and Lapata, 2018; Pappas and Popescu-Belis, 2014) have applied MIL to sentiment analysis. Angelidis and Lapata (2018) proposed a Multiple Instance Learning Network (MILNET), where the overarching polarity of a text is an aggregation of sentence or elementary discourse unit polarities, weighted by their importance. An attention-based polarity scoring method is used to obtain the importance of segments. Similar to MILNET, our model also uses an attention mechanism to obtain the importance of instances. However, the attention in our model is learned from the ACD task, while the attention in MILNET is learned from the sentiment classification task. Pappas and Popescu-Belis (2014) applied MIL to another subtask of ABSA. They proposed a multiple instance regression (MIR) model to assign sentiment scores to specific aspects of products. However, i) their task is different from ours, and ii) their model is not a neural network.

3 Model

In this section, we describe how to apply the multi-instance multi-label learning framework to the aspect-category sentiment analysis task. We first introduce the problem formulation, then describe our proposed Multi-Instance Multi-Label Learning Network for Aspect-Category sentiment analysis (AC-MIMLLN).

3.1 Problem Formulation

In the ACSA task, there are N predefined aspect categories $A = \{a_1, a_2, \dots, a_N\}$ and a predefined set of sentiment polarities $P = \{Neg, Neu, Pos\}$ (i.e., Negative, Neutral and Positive respectively). Given a sentence, $S = \{w_1, w_2, \dots, w_n\}$ and the K aspect categories, $A^S = \{A_1^S, A_2^S, \dots, A_K^S\}$, $A^S \subset A$, mentioned in S , the ACSA task predicts the sentiment polarity distributions of the K aspect categories, $p = \{p_1, p_2, \dots, p_K\}$, where $p_k = \{p_{k_{Neg}}, p_{k_{Neu}}, p_{k_{Pos}}\}$. The multi-instance multi-label learning assumes that, for the k -th aspect category, p_k is an unknown function of the unobserved word-level sentiment distributions. AC-MIMLLN first produces a sentiment distribution

p^j for each word and then combines these into a sentence-level prediction:

$$p^j = \hat{f}_{\theta_w}(w_j) \quad (1)$$

$$p_k = \hat{g}_{\theta_S}^k(p^1, p^2, \dots, p^n) \quad (2)$$

3.2 Multi-Instance Multi-Label Learning Network for ACSA

In this section, we introduce our proposed Multi-Instance Multi-Label Learning Network for Aspect-Category sentiment analysis (AC-MIMLLN), which is based on the intuitive assumption that the sentiment of an aspect category mentioned in a sentence is an aggregation of the sentiments of the words indicating the aspect category. In MIMLL, the words indicating an aspect category are called the key instances of the aspect category. Specifically, AC-MIMLLN contains two parts, an attention-based aspect category detection (ACD) classifier and an aspect-category sentiment analysis (ACSA) classifier. Given a sentence, the ACD classifier as an auxiliary task generates the weights of the words for every aspect category. The weights indicate the probabilities of the words being the key instances of aspect categories. The ACSA classifier first predicts the sentiments of the words, then obtains the sentence-level sentiment for each aspect category by combining the corresponding weights and the sentiments of the words. The overall model architecture is illustrated in Figure 2. While the ACD part contains four modules: embedding layer, LSTM layer, attention layer and aspect category prediction layer, the ACSA part also consists of four components: embedding layer, multi-layer Bi-LSTM, word sentiment prediction layer and aspect category sentiment prediction layer. In the ACD task, all aspect categories share the embedding layer and the LSTM layer, and have different attention layers and aspect category prediction layers. In the ACSA task, all aspect categories share the embedding layer, the multi-layer Bi-LSTM, and the word sentiment prediction layer, and have different aspect category sentiment prediction layers.

Input: The input of our model is a sentence consisting of n words $S = \{w_1, w_2, \dots, w_n\}$.

Embedding Layer for ACD: The input of this layer is the sentence. With an embedding matrix W_w , the sentence is converted to a sequence of vectors $X^D = \{x_1^D, x_2^D, \dots, x_n^D\}$, where $W_w \in R^{d \times |V|}$, d is the dimension of the word embeddings, and $|V|$ is the vocabulary size.

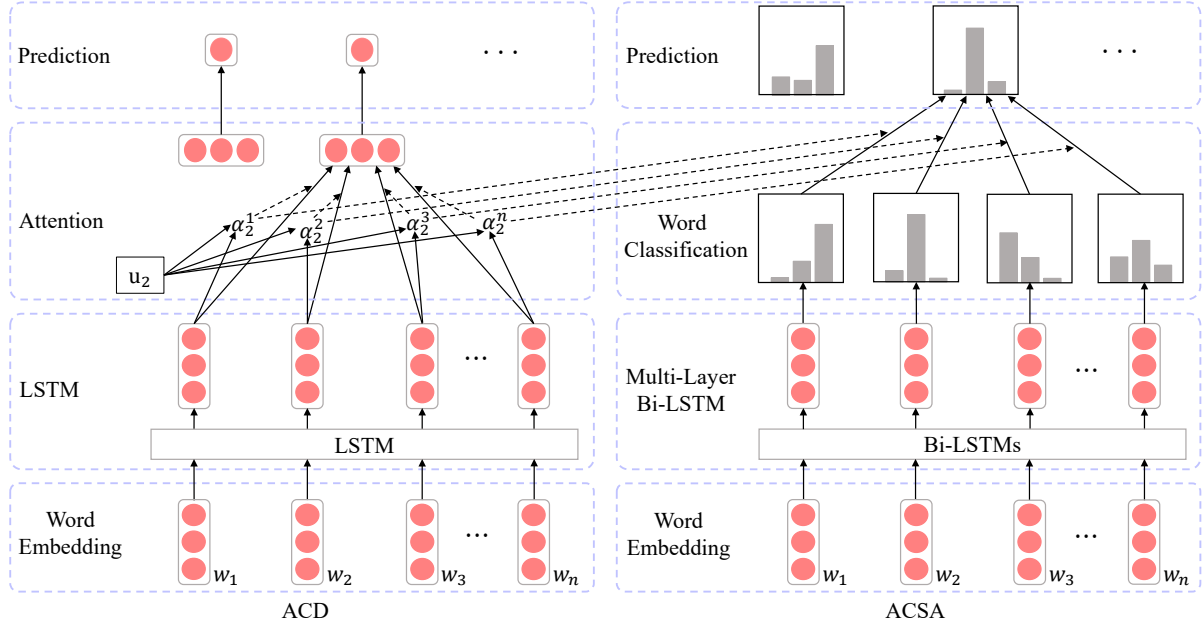


Figure 2: Overall architecture of the proposed method.

LSTM Layer: When LSTM (Hochreiter and Schmidhuber, 1997) is effective enough, attention mechanisms may not offer effective weight vectors (Wiegrefe and Pinter, 2019). In order to guarantee the effectiveness of the weights offered by attention mechanisms, we use a single-layer single-direction LSTM for ACD. This LSTM layer takes the word embeddings of the ACD task as input, and outputs hidden states $H = \{h_1, h_2, \dots, h_n\}$. At each time step i , the hidden state h_i is computed by:

$$h_i = LSTM(h_{i-1}, x_i^D) \quad (3)$$

The size of the hidden state is also set to be d .

Attention Layer: This layer takes the output of the LSTM layer as input, and produce an attention (Yang et al., 2016) weight vector for each predefined aspect category. For the j -th aspect category:

$$M_j = \tanh(W_j H + b_j), j = 1, 2, \dots, N \quad (4)$$

$$\alpha_j = \text{softmax}(u_j^T M_j), j = 1, 2, \dots, N \quad (5)$$

where $W_j \in R^{d \times d}, b_j \in R^d, u_j \in R^d$ are learnable parameters, and $\alpha_j \in R^n$ is the attention weight vector.

Aspect Category Prediction Layer: We use the weighted hidden state as the sentence representation for ACD prediction. For the j -th category:

$$r_j = H \alpha_j^T, j = 1, 2, \dots, N \quad (6)$$

$$\hat{y}_j = \text{sigmoid}(W_j r_j + b_j), j = 1, 2, \dots, N \quad (7)$$

where $W_j \in R^{d \times 1}$ and b_j is a scalar.

Embedding Layer for ACSA: For ease of reference, we use different embedding layers for ACD and ACSA. This embedding layer converts the sentence S to a sequence of vectors $X^C = \{x_1^C, x_2^C, \dots, x_n^C\}$ with the help of the embedding matrix W_w .

Multi-Layer Bi-LSTM: The output of the embedding layer for ACSA are fed into a multi-layer Bidirectional LSTM (Graves et al., 2013) (Bi-LSTM). Each layer takes the output of the previous layer as input. Formally, given the hidden states of the $(l-1)$ -th layer, $H^{l-1} = \{h_1^{l-1}, h_2^{l-1}, \dots, h_n^{l-1}\}$, the l -th Bi-LSTM outputs hidden states $H^l = \{h_1^l, h_2^l, \dots, h_n^l\}$. At each time step i , the hidden state h_i^l is computed by:

$$\vec{h}_i^l = \overrightarrow{LSTM}(\vec{h}_{i-1}^l, h_i^{l-1}) \quad (8)$$

$$\overleftarrow{h}_i^l = \overleftarrow{LSTM}(\overleftarrow{h}_{i+1}^l, h_i^{l-1}) \quad (9)$$

$$h_i^l = [\vec{h}_i^l; \overleftarrow{h}_i^l] \quad (10)$$

where $H^0 = \{x_1^C, x_2^C, \dots, x_n^C\}$, $\vec{h}_i^l \in R^{d/2}, \overleftarrow{h}_i^l \in R^{d/2}$, $h_i \in R^d$, and $d/2$ denote the size of the hidden state of LSTM. The total number of Bi-LSTM layers is L .

Word Sentiment Prediction Layer: We use the hidden state h_i^L at the time step i of the L -th layer Bi-LSTM as the representation of the i -th word, and two fully connected layers are used to produce

the i -th word sentiment prediction p^i :

$$p^i = W^2 \text{ReLU}(W^1 h_i^L + b^1) + b^2 \quad (11)$$

where $W^1 \in R^{d \times d}$, $W^2 \in R^{d \times 3}$, $b^1 \in R^d$, $b^2 \in R^3$ are learnable parameters. Note there is no softmax activation function after the fully connected layer, which lead it difficult to train our model.

Aspect Category Sentiment Prediction Layer:

We obtain the aspect category sentiment predictions by aggregating the word sentiment predictions based on the weights offered by the ACD task. Formally, for the j -th aspect category, its sentiment p_j can be computed by:

$$p_j = \text{softmax}\left(\sum_{i=1}^n p^i \alpha_j^i\right) \quad (12)$$

where $p_j \in R^3$, and α_j^i indicates the weight of the i -th word about the j -th aspect category from the weight vector α_j offered by the ACD task.

Loss: For the ACD task², as each prediction is a binary classification problem, the loss function is defined by:

$$L_A(\theta_A) = -\sum_{j=1}^N y_j \log \hat{y}_j + (1 - y_j) \log(1 - \hat{y}_j) \quad (13)$$

For the ACSA task, only the loss of the K aspect categories mentioned in the sentence is included, and the loss function is defined by:

$$L_S(\theta_S) = -\sum_{j=1}^K \sum_{c \in P} y_{jc} \log p_{jc} \quad (14)$$

We jointly train our model for the two tasks. The parameters in our model are then trained by minimizing the combined loss function:

$$L(\theta) = L_A(\theta_A) + \beta L_S(\theta_S) + \lambda \|\theta\|_2^2 \quad (15)$$

where β is the weight of ACSA loss, λ is the L_2 regularization factor and θ contains all parameters of our model.

²ACD is an auxiliary task. Although AC-MIMLLN performs both ACD and ACSA, the aspect categories it detects (i.e., the results of ACD) are usually ignored in both training stage and testing stage. The reason is that our ACD classifier is simple, it can produce effective attention weights, but may not generate effective predictions for the ACD task. In this paper, we focus on ACSA and only evaluate the performance of AC-MIMLLN on ACSA.

| Dataset | | Pos. | Neg. | Neu. |
|-------------|-------|------|------|------|
| Rest14 | Train | 1855 | 733 | 430 |
| | Dev | 324 | 106 | 70 |
| | Test | 657 | 222 | 94 |
| Rest14-hard | Test | 21 | 20 | 12 |
| MAMS-ACSA | Train | 1929 | 2084 | 3077 |
| | Dev | 241 | 259 | 388 |
| | Test | 245 | 263 | 393 |

Table 1: Statistics of the datasets.

4 Experiments

4.1 Datasets

Rest14: The SemEval-2014 restaurant review (Rest14) (Pontiki et al., 2014) dataset has been widely used. Following previous works (Cheng et al., 2017; Tay et al., 2018; Hu et al., 2019), we remove samples with conflict polarities. Since there is no official development set for Rest14, we use the split offered by Tay et al. (2018).

Rest14-hard: Following Xue and Li (2018), we construct Rest14-hard. In Rest14-hard, training set and development set are same as Rest14’s, while test set is constructed from the test set of Rest14. The test set of Rest14-hard only includes sentences containing at least two aspect categories with different sentiment polaritiess.

MAMS-ACSA: Since the test set of Rest14-hard is small, we also adopt the Multi-Aspect Multi-Sentiment dataset for Aspect Category Sentiment Analysis (denoted by MAMS-ACSA). MAMS-ACSA is released by Jiang et al. (2019), all sentences in which contain multiple aspect categories with different sentiment polarities.

We select Rest14-hard and MAMS-ACSA that we call hard datasets because most sentences in Rest14 contain only one aspect or multiple aspects with the same sentiment polarity, which makes ACSA degenerate to sentence-level sentiment analysis (Jiang et al., 2019). Rest14-hard and MAMS-ACSA can measure the ability of a model to detect multiple different sentiment polarities in one sentence toward different aspect categories. Statistics of these three datasets are given in Table 1.

4.2 Comparison Methods

We compare AC-MIMLLN with various baselines. (1) non-BERT models: GCAE (Xue and Li, 2018),

| Methods | Rest14 | Rest14-hard | MAMS-ACSA |
|-----------------------------------|--------------------------------------|--------------------------------------|--------------------------------------|
| GCAE (Xue and Li, 2018) | 81.336(\pm 0.883) | 54.717(\pm 4.920) | 72.098 \dagger |
| As-capsule (Wang et al., 2019) | 82.179(\pm0.414) | 60.755(\pm 2.773) | 75.116(\pm 0.473) |
| CapsNet (Jiang et al., 2019) | 81.172(\pm 0.631) | 53.962(\pm 0.924) | 73.986 \dagger |
| AC-MIMLLN (ours) | 81.603(\pm 0.715) | 65.283(\pm2.264) | 76.427(\pm0.704) |
| AC-MIMLLN – w/o mil (ours) | 80.596(\pm 0.816) | 64.528(\pm 2.201) | 75.650(\pm 1.100) |
| AC-MIMLLN-Affine (ours) | 80.843(\pm 0.760) | 64.151(\pm 3.375) | 74.517(\pm 1.299) |
| BERT (Jiang et al., 2019) | 87.482(\pm 0.906) | 67.547(\pm 5.894) | 78.292 \dagger |
| BERT-pair-QA-B (Sun et al., 2019) | 87.523(\pm 1.175) | 69.433(\pm 4.368) | 79.134(\pm 0.973) |
| CapsNet-BERT (Jiang et al., 2019) | 86.557(\pm 0.943) | 51.321(\pm 1.412) | 79.461 \dagger |
| AC-MIMLLN-BERT (ours) | 89.250(\pm0.720) | 74.717(\pm3.290) | 81.198(\pm0.606) |

Table 2: Results of the ACSA task in terms of accuracy (% , mean \pm (std)). \dagger refers to citing from Jiang et al. (2019).

| Methods | food | ser-vice | amb-ience | price | misc |
|------------|-------------|-------------|-------------|-------------|-------------|
| As-capsule | 82.7 | 90.1 | 84.3 | 80.5 | 74.6 |
| AC-MIMLLN | 83.7 | 90.5 | 83.6 | 84.0 | 69.0 |

Table 3: Results of the ACSA task on Rest14’s aspect categories in terms of accuracy (%).

As-capsule (Wang et al., 2019)³ and CapsNet (Jiang et al., 2019); (2) BERT (Devlin et al., 2019) based models: BERT (Jiang et al., 2019), BERT-pair-QA-B (Sun et al., 2019) and CapsNet-BERT (Jiang et al., 2019). We also provide the comparisons of several variants of AC-MIMLLN:

AC-MIMLLN – w/o mil generates aspect category-specific representations for the ACAC task. The representations are the weighted sum of the word representations based on the weights offered by the ACD task.

AC-MIMLLN-Affine replaces the LSTM in AC-MIMLLN with an affine hidden layer, which is used to evaluate the effectiveness of the attention in AC-MIMLLN (Wiegrefe and Pinter, 2019).

AC-MIMLLN-BERT replaces the embedding layer for ACSA and the multi-layer Bi-LSTM in AC-MIMLLN with the uncased basic pre-trained BERT. Since the overall sentiment of a sentence as context information is important for inferring the sentiment of a particular aspect category, AC-MIMLLN-BERT also predicts the sentiment of the token “[CLS]” and assigns weight 1 to it. AC-MIMLLN-BERT takes “[CLS] sentence [SEP] aspect category [SEP]” as input like CapsNet-BERT.

³As-capsule is also a multi-task model, which performs ACD and ACSA simultaneously like our model.

4.3 Implementation Details

We implement our models in PyTorch (Paszke et al., 2017). We use 300-dimensional word vectors pre-trained by GloVe (Pennington et al., 2014) to initialize the word embedding vectors. The batch sizes are set to 32 and 64 for non-BERT models on the Rest14(-hard) dataset and the MAMS-ACSA dataset, respectively, and 16 for BERT-based models. All models are optimized by the Adam optimizer (Kingma and Ba, 2014). The learning rates are set to 0.001 and 0.00002 for non-BERT models and BERT-based models, respectively. We set $L = 3$, $\lambda = 0.00001$ and $\beta = 1$. For the ACSA task, we apply a dropout of $p = 0.5$ after the embedding and Bi-LSTM layers. For AC-MIMLLN-BERT, ACD is trained first then both of ACD and ACSA are trained together. For other models, ACD and ACSA are directly trained jointly. We apply early stopping in training and the patience is 10. We run all models for 5 times and report the average results on the test datasets.

4.4 Experimental Results

Experimental results are illustrated in Table 2. According to the experimental results, we can come to the following conclusions. First, AC-MIMLLN outperforms all non-BERT baselines on the Rest14-hard dataset and the MAMS-ACSA dataset, which indicates that AC-MIMLLN has better ability to detect multiple different sentiment polarities in one sentence toward different aspect categories. Second, AC-MIMLLN obtains +1.0% higher accuracy than AC-MIMLLN – w/o mil on the Rest14 dataset, +0.8% higher accuracy on the Rest14-hard dataset and +0.8% higher accuracy on the MAMS-ACSA dataset, which shows that the Multiple Instance Learning (MIL) framework is more suitable for

| Methods | Rest14 | | Rest14-hard | | MAMS-ACSA | |
|------------------|---------------|---------------|---------------|---------------|---------------|---------------|
| | KID(F_1) | KISC(acc) | KID(F_1) | KISC(acc) | KID(F_1) | KISC(acc) |
| AC-MIMLLN | 38.132 | 73.731 | 48.927 | 62.250 | 69.480 | 68.804 |
| AC-MIMLLN-Affine | 60.035 | 76.503 | 66.920 | 67.250 | 75.083 | 68.503 |
| AC-MIMLLN-BERT | 63.477 | 83.894 | 70.027 | 72.250 | 74.172 | 75.666 |

Table 4: Performance of detecting the key instances (KID) of the given aspect category in terms of accuracy (%) and classifying the sentiments of the given key instances (KISC) in terms of F_1 measure (%).

| Methods | Rest14 | Rest14-hard | MAMS-ACSA |
|-----------------|---------------|---------------|---------------|
| single-pipeline | 81.459 | 61.509 | 72.231 |
| single-joint | 80.329 | 62.641 | 75.605 |
| multi-pipeline | 82.117 | 63.396 | 72.675 |
| multi-joint | 81.603 | 65.283 | 76.427 |

Table 5: Results of AC-MIMLLN in different multi-task settings on ACSA in terms of accuracy(%).

the ACSA task. Third, AC-MIMLLN-BERT surpasses all BERT-based models on all three datasets, indicating that AC-MIMLLN can achieve better performance by using more powerful sentence encoders for ACSA. In addition, AC-MIMLLN can't outperform As-capsule on Rest14. The main reason is that AC-MIMLLN has poor performance on the aspect category *misc* (the abbreviation for *anecdotes/miscellaneous*) (see Table 3 and Figure 4 (f)).

4.5 Impact of Multi-Task Learning

AC-MIMLLN is a multi-task model, which performs ACD and ACSA simultaneously. Multi-task learning (Caruana, 1997) achieves improved performance by exploiting commonalities and differences across tasks. In this section, we explore the performance of AC-MIMLLN in different multi-task settings on the ACSA task. Specifically, we explore four settings: single-pipeline, single-joint, multi-pipeline and multi-joint. The "single" means that the ACSA task predicts the sentiment of one aspect category in sentences every time, while the "multi" means that the ACSA task predicts the sentiments of all aspect categories in sentences every time. The "pipeline" indicates that ACD is trained first, then ACSA is trained, while the "joint" indicates ACD and ACSA are trained jointly. The multi-joint is AC-MIMLLN.

Experimental results are shown in Table 5. First, we observe that, multi-* outperform all their counterparts, indicating modeling all aspect categories in sentences simultaneously can improve the per-

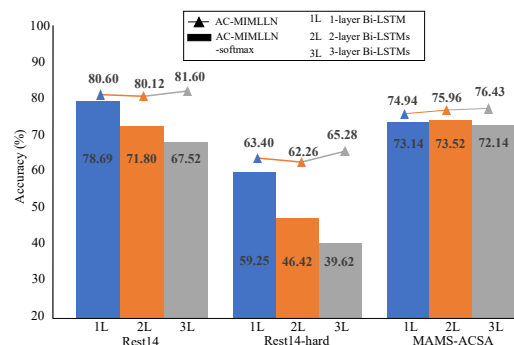


Figure 3: The impact of the number of Bi-LSTM layers and the softmax activation function.

formance of the ACSA task. Second, *-joint surpass *-pipeline on the Rest14-hard dataset and the MAMS-ACSA dataset, which shows that training ACD and ACSA jointly can improve the performance on hard datasets. Third, *-joint obtain worse performance on the Rest14 dataset than *-pipeline. One possible reason is that Rest14 is simple and *-joint have bigger model capacity than *-pipeline and overfit on Rest14.

4.6 Impact of Multi-layer Bi-LSTM Depth

In this section, we explore the effect of the number of the Bi-LSTM layers. Experiments results are shown in Figure 3, which also contains the results of AC-MIMLLN-softmax. AC-MIMLLN-softmax is obtained by adding the softmax activation function to the word sentiment prediction layer of AC-MIMLLN. We observe that, when the number of Bi-LSTM layer increases, AC-MIMLLN usually obtains better performance, and AC-MIMLLN-softmax obtains worse results. It indicates that AC-MIMLLN-softmax is hard to train when its complexity increases, while AC-MIMLLN can achieve better performance by using more powerful sentence encoders for ACSA.

4.7 Quality Analysis

In this subsection, we show the advantages of our model and analyze where the error lies in through

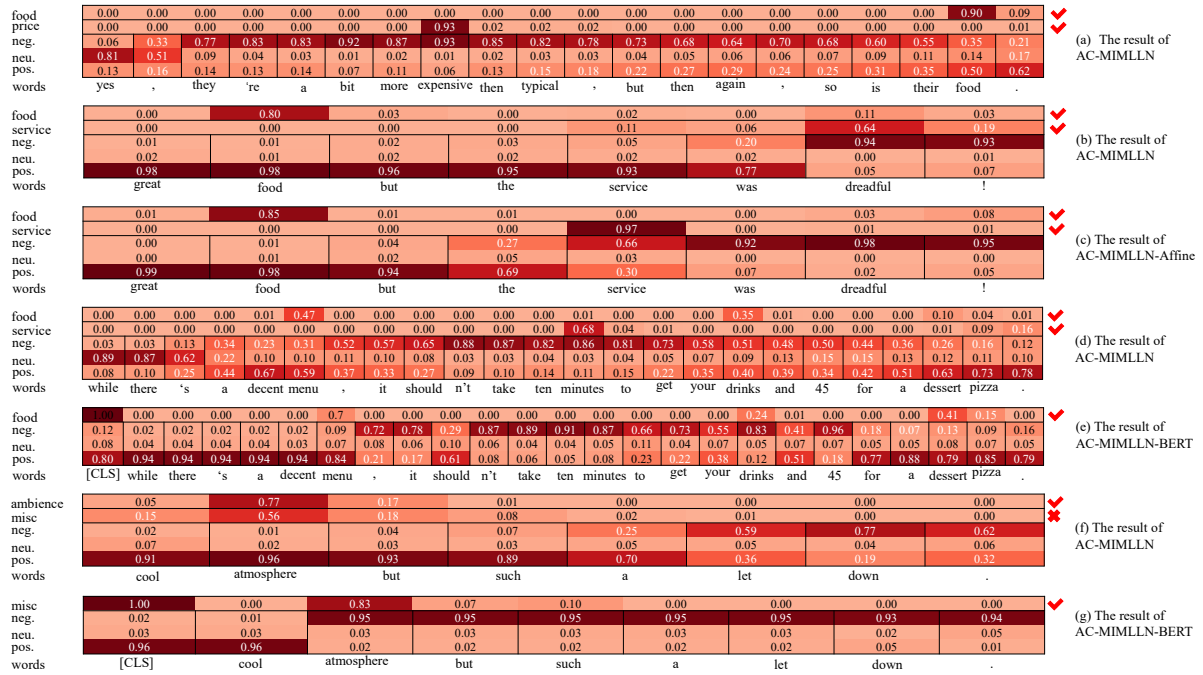


Figure 4: Visualization of the attention weights and the word sentiment prediction results. For each subfigure, the lines corresponding aspect categories show the attention weights offered by the ACD task, while the other three lines show the word sentiment distributions predicted by the ACSA task.

some typical examples and estimating the performance of our model detecting the key instances (KID) of the given aspect category and classifying the sentiments of the given key instances (KISC). We annotate the key instances for the aspect categories mentioned in sentences and their sentiment polarities on the test set of the three datasets. Models judge a word as a key instance if the weight of the word is greater than or equal to 0.1. Experimental results are illustrated in Table 4.

Case Study Figure 4 visualizes the attention weights and the word sentiment prediction results of four sentences. Figure 4 (a) shows that, our model accurately finds the key instances “expensive” for the aspect category *price* and “food” for *food*, and assigns correct sentiments to both the aspect categories and the key instances. Compared with previous models, which generate aspect category-specific sentence representations for the ACSA task directly (e.g. BERT-pair-QA-B) or based on aspect category-related sentiment words (e.g. As-capsule), our model is more interpretable.

In Figure 4, (b) and (c) show that, both AC-MIMLLN and AC-MIMLLN-Affine can correctly predict the sentiments of the aspect categories, *food* and *service*. While AC-MIMLLN-Affine accurately find the key instance “service” for *service*, AC-MIMLLN assigns weights to all the words

in the text snippet “service was dreadful!”. This is because the LSTM-based ACD model in AC-MIMLLN can select useful words for both ACD and ACSA based on the context, which results in better performance (see Table 2). This also can explain why AC-MIMLLN has worse performance on detecting the key instances of the given aspect category than AC-MIMLLN-Affine (see Table 4).

Error Analysis In Figure 4 (d), the sentiments toward “drinks” and “dessert” (key instances of the aspect category *food*) should be neutral, however AC-MIMLLN assigns negative sentiment to “drinks” and positive sentiment to “dessert”. Figure 4 (e) shows AC-MIMLLN-BERT also assigns wrong sentiments to “drinks” and “dessert”. Table 4 shows that although AC-MIMLLN-BERT significantly improve the performance of KISC, its results are also less than 80% on the Rest14-hard dataset and the MAMS-ACSA dataset.

In Figure 4 (f), AC-MIMLLN wrongly predict the sentiment of the aspect category *misc*, because it finds the wrong key instances for *misc*. Compared to other aspect categories, it’s harder to decide which words are the key instances of *misc* for AC-MIMLLN, resulting in poor performance of AC-MIMLLN on the aspect category *misc*. Figure 4 (g) shows AC-MIMLLN-BERT correctly predict the sentiments of the aspect category *misc*, but

also finds the wrong key instances for *misc*. Table 4 shows that all results on KID are less than 75%.

5 Conclusion

In this paper, we propose a Multi-Instance Multi-Label Learning Network for Aspect-Category sentiment analysis (AC-MIMLLN). AC-MIMLLN predicts the sentiment of an aspect category mentioned in a sentence by aggregating the sentiments of the words indicating the aspect category in the sentence. Experimental results demonstrate the effectiveness of AC-MIMLLN. Since AC-MIMLLN finds the key instances for the given aspect category and predicts the sentiments of the key instances, it is more interpretable. In some sentences, phrases or clauses rather than words indicate the given aspect category, future work could consider multi-grained instances, including words, phrases and clauses. Since directly finding the key instances for some aspect categories is ineffective, we will try to first recognize all opinion snippets in a sentence, then assign these snippets to the aspect categories mentioned in the sentence.

References

- Stefanos Angelidis and Mirella Lapata. 2018. [Multi-instance learning networks for fine-grained sentiment analysis](#). *Transactions of the Association for Computational Linguistics*, 6:17–31.
- Rich Caruana. 1997. [Multitask learning](#). *Machine learning*, 28(1):41–75.
- Zenghai Chen, Zheru Chi, Hong Fu, and Dagan Feng. 2013. [Multi-instance multi-label image classification: A neural approach](#). *Neurocomputing*, 99:298–306.
- Jiajun Cheng, Shenglin Zhao, Jiani Zhang, Irwin King, Xin Zhang, and Hui Wang. 2017. [Aspect-level sentiment classification with heat \(hierarchical attention\) network](#). In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 97–106.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [Bert: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.
- Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. [Speech recognition with deep recurrent neural networks](#). In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. [Long short-term memory](#). *Neural computation*, 9(8):1735–1780.
- Mengting Hu, Shiwan Zhao, Li Zhang, Keke Cai, Zhong Su, Renhong Cheng, and Xiaowei Shen. 2019. [Can: Constrained attention networks for multi-aspect sentiment analysis](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4593–4602.
- Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. [A challenge dataset and effective models for aspect-based sentiment analysis](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6281–6286.
- Xiaotian Jiang, Quan Wang, Peng Li, and Bin Wang. 2016. [Relation extraction with multi-instance multi-label convolutional neural networks](#). In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1471–1480.
- Jim Keeler and David E Rumelhart. 1992. A self-organizing integrated segmentation and recognition neural net. In *Advances in neural information processing systems*, pages 496–503.
- Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Dimitrios Kotzias, Misha Denil, Nando De Freitas, and Padhraic Smyth. 2015. From group to individual labels using deep features. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 597–606.
- Zeyang Lei, Yujiu Yang, Min Yang, Wei Zhao, Jun Guo, and Yi Liu. 2019. [A human-like semantic cognition network for aspect-level sentiment classification](#). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6650–6657.
- Cheng Li, Xiaoxiao Guo, and Qiaozhu Mei. 2017. [Deep memory networks for attitude identification](#). In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 671–680.
- Yuncong Li, Cunxiang Yin, T. Wei, Huiqiang Zhong, Jinchang Luo, Siqi Xu, and Xiaohui Wu. 2019. A joint model for aspect-category sentiment analysis with contextualized aspect embedding. *ArXiv*, abs/1908.11017.

- Yunlong Liang, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2019. A novel aspect-guided deep transition model for aspect based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5572–5584.
- Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.
- Guoqing Liu, Jianxin Wu, and Zhi-Hua Zhou. 2012. Key instance detection in multi-instance learning.
- Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135.
- Nikolaos Pappas and Andrei Popescu-Belis. 2014. Explaining the stars: Weighted multiple-instance learning for aspect-based sentiment analysis. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 455–466.
- Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.
- Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, San Diego, California. Association for Computational Linguistics.
- Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, Denver, Colorado. Association for Computational Linguistics.
- Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland. Association for Computational Linguistics.
- Sebastian Ruder, Parsa Ghaffari, and John G. Breslin. 2016. A hierarchical model of reviews for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 999–1005, Austin, Texas. Association for Computational Linguistics.
- Martin Schmitt, Simon Steinheber, Konrad Schreiber, and Benjamin Roth. 2018. Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1109–1114.
- Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 380–385.
- Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 455–465. Association for Computational Linguistics.
- Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615.
- Yequan Wang, Aixin Sun, Minlie Huang, and Xiaoyan Zhu. 2019. Aspect-level sentiment analysis using as-capsules. In *The World Wide Web Conference*, pages 2033–2044.
- Sarah Wiegrefe and Yuval Pinter. 2019. Attention is not not explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 11–20.
- Bowen Xing, Lejian Liao, Dandan Song, Jingang Wang, Fuzheng Zhang, Zhongyuan Wang, and Heyan Huang. 2019. Earlier attention? aspect-aware lstm for aspect sentiment analysis. *arXiv preprint arXiv:1905.07719*.
- Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2514–2523.

- Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. [Hierarchical attention networks for document classification](#). In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.
- Min-Ling Zhang and Zhi-Hua Zhou. 2008. M3miml: A maximum margin method for multi-instance multi-label learning. In *2008 Eighth IEEE International Conference on Data Mining*, pages 688–697. IEEE.
- Zhi-Hua Zhou and Min-Ling Zhang. 2006. Multi-instance multi-label learning with application to scene classification. In *Proceedings of the 19th International Conference on Neural Information Processing Systems*, pages 1609–1616. MIT Press.
- Peisong Zhu, Zhuang Chen, Haojie Zheng, and Tiejun Qian. 2019. [Aspect aware learning for aspect category sentiment analysis](#). *ACM Trans. Knowl. Discov. Data*, 13(6).