Multi-label Aspect Classification on Question-Answering Text with Contextualized Attention-based Neural Network

Hanqian Wu[1,2⋆], Shangbin Zhang[1,2], Jingjing Wang[3,], Mumu Liu[1,2], and Shoushan Li[3]

[1] School of Computer Science and Engineering, Southeast University, China
[2] Key Laboratory of Computer Network and Information Integration of Ministry of Education, Southeast University, China
[3] NLP Lab, School of Computer Science and Technology, Soochow University, China
hanqian@seu.edu.cn, ternencewind@outlook.com, djingwang@gmail.com, liudoublemu@163.com, lishoushan@suda.edu.cn

Abstract. In the e-commerce websites, such as Taobao and Amazon, interactive question-answering (QA) style reviews usually carry rich aspect information of products. To well automatically analyze the aspect information inside QA style reviews, it' s worthwhile to perform aspect classification on them. Unfortunately, until now, there are few papers that focus on performing aspect classification on the QA style reviews. For short, we referred to this novel task as QA aspect classification (QA-AC). In this study, we model this task as a multi-label classification problem where each QA style review is explicitly mapped to multiple aspect categories instead of only one aspect category. To solve this issue, we propose a contextualized attention-based neural network approach to capture both the contextual information and the QA matching information inside QA style reviews for the task of QA-AC. Specifically, we first propose two aggregating strategies to integrate multi-layer contextualized word embeddings of the pre-trained language representation model (i.e., BERT) so as to capture contextual information. Second, we propose a bidirectional attention layer to capture the QA matching information. Experimental results demonstrate the effectiveness of our approach to QA-AC.

Keywords: Aspect Classification · Question Answering · Pre-trained Language Model · Bidirectional Attention

1 Introduction

In recent years, a new form of the product review, called Question-Answering (QA) style review, has emerged on some e-commerce platforms, including Taobao, Yelp, and Amazon. As shown in Fig 1, unlike traditional product reviews, a QA style review consists of a question sentence and an answer sentence. Thus, we

⋆ Corresponding Author

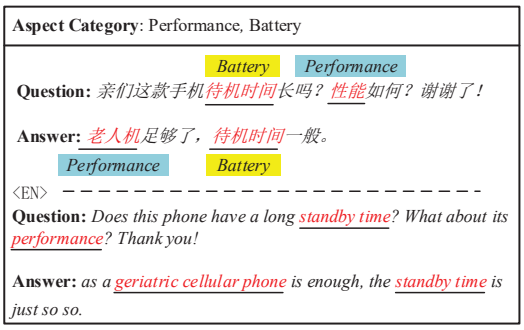

Fig. 1. A translated example of QA style reviews from an e-commerce website.

also regard QA style reviews as QA pairs. QA aspect classification (QA-AC) is an essential fundamental task in sentiment analysis for e-commerce reviews, which aims to identify the aspect set of the product contained in a QA style review. The customary e-commerce product review is mainly written for subjectively and generally commenting after the transaction completed, which may not answer questions from other consumers. Thus, QA style reviews become widespread and proliferating. Notably, due to the nature of the conversation, questions often focus on some aspects of the product, where answers also aim at. Therefore, QA style reviews are more suitable for aspect-based sentiment analysis tasks, and aspect-based sentiment analysis research on QA style reviews is capturing increasing attention. Aspect classification is an essential sub-task of the aspect-based sentiment analysis task, and it can improve the performance of aspect-based sentiment analysis significantly. However, most of the relevant researches are based on traditional reviews, while few studies focus on the task of QA-AC. In general, the QA-AC exists the following three specific challenges.

First, instead of one individual aspect, the aspect categories of the sample shown in Fig 1 include *Performance* and *Battery*. It is ubiquitous in QA style reviews that a single review involves more than one aspect. However, classifying a review into multiple aspect labels have more difficulties than classifying a review into a single label. In this study, we regard the QA-AC task as a multi-label classification problem and accommodate our model to this task.

Second, a QA style review is a short text, it's rather difficult to identify the multiple aspect categories inside it due to data sparseness problem. Conventional language representation models perform poorly in obtaining semantic information from short texts. In addition, informal expressions are widespread in QA style reviews, and existing Chinese words segmentation utilities could not correctly recognize some informal expressions. For instance, in Fig 1, the phrase "老人机 (geriatric cellular phone)" means that this phone does not have many complex functions and excellent performance. After the word segmentation, the phrase will split to "老人 (elder people)" and "机 (machine)". Obviously, according to this instance, if using word segmentation, it's difficult to capture the performance information of the phone. To address these problems, we adopt the latest outstanding pre-trained language representation model, i.e., BERT, which

can effectively alleviate the data sparseness and polysemy problem by modeling the context semantic representations with large-scale external data.

Third, the QA style review has much contextual matching information between its question part and answer part. In QA style reviews, the matching information between question and answer plays a crucial role in QA-AC task. For example, the phrase "待机时间 (standby time)" in the answer part has a strong correlation with the same phrase in question part than other words. Its correlation provides valuable clues to predict the aspect of this review to *Battery*. Therefore, we construct the bidirectional attention neural network model to detect the important degrees of different characters.

In this paper, with the best of our knowledge, we are the first to define the QA-AC task as a multi-label classification problem, which aims to identify multiple aspect categories inside a given QA style review. Furthermore, we propose a contextualized attention-based Network, i.e., the Bidirectional Attention Neural Network (BANN) based on BERT, for capturing both the contextual information and the QA matching information. In detail, we first adopt BERT as a contextual embedding to alleviate the data sparseness and polysemy problem. Specifically, we employ two strategies for aggregating multi-layer pre-trained semantic representations so as to capture contextual information, including averaging and weighted summation. Then, we encode QA matching information via a bidirectional QA matching attention. Finally, the empirical results demonstrate that our proposed model outperforms several state-of-the-art baselines by larger margins on QA style reviews.

2 Related Work

2.1 Aspect Extraction

Over the last decade, as a fine-grained sentiment analysis task, aspect-level sentiment classification captured enormous attention, especially in the field of reviews texts [1–3]. As a related task to the aspect-level sentiment classification, aspect extraction divide into two sub-tasks, aspect term extraction and aspect category classification, which is also called aspect classification. In the beginning, researchers proposed several rule-based methods and traditional machine learning methods for aspect extraction. For example, Rubtsova et al. [4] used the conditional random field method to extract the aspect term mentioned in the restaurants and automobiles texts. With the prosperity of deep learning, the neural networks based methods are widely employed in natural language processing tasks and remarkably adept in learning complicated feature representations automatically. Liu et al. [5] proposed a Convolutional Neural Network (CNN) model employed two types of pre-trained embeddings for aspect extraction, including general-purpose embedding and domain-specific embedding. However, they focus on general-purpose information and domain information in the embedding layer while ignoring the different importance between two embeddings. He et al. [6] proposed an attention-based model intending to discover coherent aspects and proved the efficiency of attention mechanism.

Different from the above, this paper explores the methods of QA-AC task. However, Wu et al. [7] are the first to conduct the research on QA-AC by impracticably assuming that a QA style review only contains a single aspect. In this paper, we first define the QA-AC task as a multi-label classification problem and design a bidirectional QA attention layer to capture the QA matching information.

2.2 Pre-trained Language Model

Recently, the language representation model has received considerable attention due to improving scores of many natural language processing (NLP) tasks [8, 9]. The pre-trained language model (PLM) is a type of deep language representation model, which is pre-trained on a large unlabelled text corpus. There are two existing strategies in the PLM. Embeddings from Language Models (ELMo) [10] is a feature-based pre-trained language model, and the Generative Pre-trained Transformer (OpenAI GPT) [11] utilizes a fine-tuning approach to apply pre-trained language representations to downstream tasks. Different from Word2vec, PLM can generate deep contextualized word representations. However, these existing PLMs cannot make good use of contextual information because they are unidirectional. Thus, Devlin et al. [12] proposed the Bidirectional Encoder Representations from Transformers (BERT). BERT addresses the mentioned unidirectional constraints and polysemy problem and obtains new state-of-the-art results on eleven NLP tasks. Thus, some researches leverage BERT to improve their models. Adhikari et al. [13] explored BERT on document classification. Their model achieves state-of-the-art across four popular datasets.

In our study, we utilize BERT to capture the contextual information in QA style reviews and leverage dominant context-dependent word representations to expand contextual information and alleviate the data sparseness and polysemy problem. To our knowledge, we are the first to leverage BERT on the QA-AC task.

3 Models

We model the QA-AC task as a multi-label classification task. For introducing our model explicitly, we formulate the task as follows:

Given a QA style review text $C = \{c_1, \cdots, c_n\}$, the target is to predict the aspect set $S = \{s_1, \cdots, s_k\}$ of C, where c_i denotes the i-th character in C which has n characters. As a QA pair, the text $C = (Q, A)$ can split into a question part and an answer part. Besides, there are k aspects that occur in the whole corpus. If the text C has the i-th aspect, its s_i should take the value of 1. As a multi-label problem, the set S may contain more than one aspect, and the value of elements in S which represent other aspects are zeros.

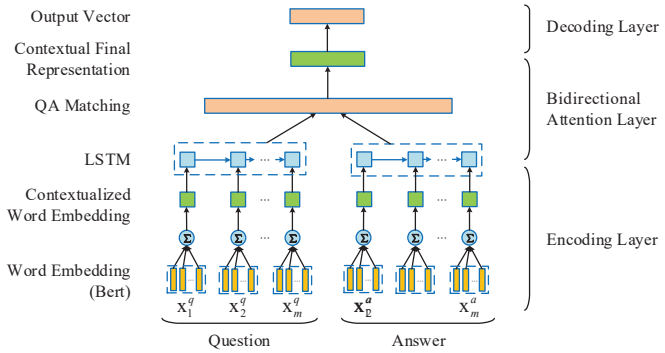

Fig. 2. The overall structure of our model.

3.1 Model Overview

Fig 2 depicts the illustration of our model architecture. In the macroscopic view, our model can decompose into three layers:

Encoding layer encodes the text C into a contextualized sentence vector which consists of word vectors. For obtaining the contextualized information in word embedding, we leverage the BERT contextual embeddings [12] to encode text C. Especially, we apply two strategies, averaging and weighted summation, to aggregate embeddings from twelve BERT transformer layers .

Bidirectional Attention layer captures the contextual matching information between question and answer by a bidirectional QA matching attention. Moreover, we conduct the Long Short-Term Memory (LSTM) to learn the order information hidden in the sentence. Hence, after the whole attention layer, we can get the contextual representation for each text.

Decoding layer compresses the representation vector's dimension and generates the output vector, which is the predicted aspect set of this text. Distinct from the multi-class classification task, we employ the sigmoid function as the activation function.

3.2 Encoding Layer

BERT Word Embedding: The first step of our model is to embed our input QA pair. BERT for Chinese is a character-level model, whose output is a vector with fixed dimensions. Firstly, after padding the question and answer to the same length m, we stitch them into a single sentence $C = \{c_1^q, c_2^q, \cdots, c_m^q, c_1^a, c_2^a, \cdots, c_m^a\}$. Then, we feed the BERT model with vector C directly. BERT generates 12 layers of hidden states for every token in total, especially, which can be all used to present words. After BERT word embedding, we get the vector $X = \{x_1^q, x_2^q, \cdots, x_q^m, x_1^a, x_2^a, \cdots, x_m^a\}$, and x_i represents $\{x_i{}^1, x_i{}^2, \cdots, x_i{}^L\}$, where $x_i{}^j$ denotes the j-th layer representation of the i-th character.

Contextualized Word Embedding: In the component of layers aggregating, we apply two strategies to aggregate embeddings from these twelve layers. The

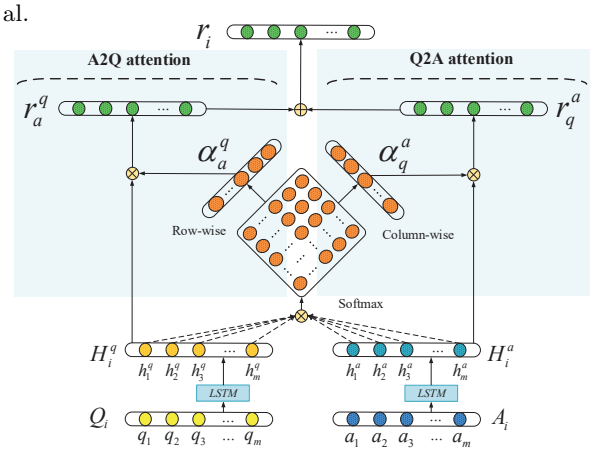

Fig. 3. The structure of Bidirectional Attention Layer.

first strategy is averaging. We compute the mean value of each x_i to encode the i-th character. The other strategy is the weighted summation shown below:

$$q_i = \sum_{j=1}^{L} \omega_j x_{ij}^q, \quad a_i = \sum_{j=1}^{L} \omega_j x_{ij}^a \tag{1}$$

In detail, q_i denotes the word vector of i-th character in question and a_i represents the word vector of i-th character in answer. ω_j is the trainable weight parameter of the j-th layer vector. We compare two strategies in our experiment part, and the results show that the weighted summation outperforms averaging.

3.3 Bidirectional Attention Layer

Undoubtedly, there is sufficient QA information hidden in QA style reviews, which can contribute to the aspect classification task significantly. Figure 3 reveals the detailed structure of the bidirectional attention layer.

LSTM: For efficiently making use of sequential internal correlation, we employ two LSTM networks to capture the contextual order information in question and answer severally. Both question $Q_i = \{q_1, q_2, \cdots, q_m\}$ and answer $A_i = \{a_1, a_2, \cdots, a_m\}$ have m elements. Therefore, we collect entire m hidden states of LSTM, $H_i^q = \{h_1^q, h_2^q, \cdots, h_m^q\}$ and $H_i^a = \{h_1^a, h_2^a, \cdots, h_m^a\}$, where h_j^q denotes the j-th hidden state in question and h_j^a denotes the j-th hidden state in answer.

QA Matching: In this component, we mine the bidirectional QA matching information. Firstly, we calculate the matching matrix by $D = H_i^q (H_i^a)^\mathsf{T}$, which denotes the bidirectional pair-wise matching information. Then, we compute two directional attention matrixes, including Question-to-Answer attention(Q2A attention) and Answer-to-Question attention(A2Q attention).

- **A2Q attention:** We employ a series of row-wise operations to obtain A2Q attention weight vector α_a^q by Equation (2).

$$\alpha_{row} = \sum_{i=0}^{m} softmax_{row-wise}(D_i), \quad \alpha_a^q = softmax(\alpha_{row}) \tag{2}$$

  where $\alpha_a^q \in \mathbb{R}^m$ is the A2Q attention vector which presents the importance of characters in question Q_i, α_{row} is the row-wise matching vector, and D_i is the i-th row in matrix D. To strengthen the features in each row, we compute the row-wise softmax before the summation operation in matrix D.
- **Q2A attention:** Different from above, we employ some column-wise operations to obtain Q2A attention weight vector α_q^a by Equation (3)

$$\alpha_{col} = \sum_{j=0}^{m} softmax_{column-wise}(D^{\mathsf{T}}{}_j), \quad \alpha_q^a = softmax(\alpha_{col}) \tag{3}$$

  where $\alpha_q^a \in \mathbb{R}^m$ is the Q2A attention vector which presents the importance of characters in answer A_i, α_{col} is the column-wise matching vector, and $D^{\mathsf{T}}{}_j$ is the j-th column in D. Similarly, we compute the column-wise softmax before the column-wise summation operation in matrix D.

Contextual Final Representation: After bidirectional attention, we can get two directional sentence representations, $r_a^q \in \mathbb{R}^d$ and $r_q^a \in \mathbb{R}^d$, where d is the dimension of BERT word vector. We combine A2Q sentence vector r_a^q and Q2A sentence vector r_q^a to present the contextualized final representation as follows:

$$r_a^q = \alpha_a^q H_i^q, \ r_q^a = \alpha_q^a H_i^a, \ r_i = W_r(r_a^q \oplus r_q^a) + B_r \tag{4}$$

where $\oplus$ denotes the concatenate operator, $r_i \in \mathbb{R}^d$ is the contextualized final representation of C_i, $W_r \in \mathbb{R}^{2d \times d}$ is the weighted matrix, and B_r is the bias matrix. After compression, we can obtain a d-dim vector.

3.4 Decoding Layer

After transforming each text C into a contextual representation, we conduct a dense layer to generate the ultimate vector of aspects in C. The whole computational process of decoding layer is shown below:

$$out_i = sigmoid(W_l r_i + B_l) \tag{5}$$

where $out_i \in \mathbb{R}^k$ is the predicted vector of text C_i and k is the number of all aspects, W_l is an intermediate weight matrix, B_l is a bias matrix. Notably, different from the multi-class classification problem, we employ sigmoid as the final activation function to evaluate the possibility of each category respectively. In reality, for cooperating with the sigmoid activation function, we set the binary cross-entropy as our objective function, which we will detailedly expatiate later.

Table 1. Statistics of our QA dataset.

Aspect category	number of all instances	the scale of multi-label instances
Quality	214	20.1%
Battery	303	23.1%
Performance	652	16.9%
Certified product	421	10.7%
IO	969	12.3%
Function	120	7.5%
Computation	126	21.4%
Total	2586	7.9%

3.5 Model training

Our model can be trained end-to-end by backpropagation. For minimizing the loss of each aspect respectively, the binary cross entropy is selected as our objective function. The Equation (6) demonstrates how the loss is calculated out.

$$loss = -\frac{1}{n}\sum_{i=0}^{k}(y[i] \times \log(\hat{y}[i]) + (1-y) \times \log(1-\hat{y}[i])) \tag{6}$$

where y is the ground truth label set for the given text and $\hat{y}$ is the prediction of our model, k is the number of total aspects, and $y[i]$ denotes that whether the given text contains the $(i+1)$-th aspect. We average all cross entropy to measure the model loss.

In our experiments, we employ Adam [14] to optimize trainable parameters in our model, which adaptively modify its learning rate during the training process.

4 Experiments

4.1 Experimental Settings

- **Datasets** We conduct our experiments on the QA style reviews dataset collected from *"asking all"* in Taobao that Wu et al. [7] provided. The whole corpus has 2580 reviews of *electronic appliances* and contains seven aspects of products. Each review may express more than one aspect, and we aim to identify the whole aspects list of each review. To better illustrate the data distribution, the statistics are reported in Table 1. Notably, the scale of multi-label instances in major aspects exceeds 10%. It means that settling the multi-label problem in QA-AC task is essential and ponderable.

- **Evaluation Metrics** In our experiments, three evaluation metrics are employed to measure the performance of each experiment, including hamming loss, accuracy, and F1-measure.
  Hamming loss is the fraction of labels that are incorrectly predicted, and the smaller hamming loss manifests a better performance.

$$hammingloss = 1 - \frac{1}{nk}\sum_{i=1}^{n}\sum_{j=1}^{k}\mathbb{I}(y_i^j = \hat{y}_i^j) \tag{7}$$

Table 2. Experimental results. The best scores are in bold.

Models	Hamming loss	Accuracy	F1-measure
Binary Relevance [16]	0.064	0.662	0.707
Classifier Chains [16]	0.032	0.836	0.886
LSTM(Word2vec) [3]	0.031	0.843	0.901
BANN(Word2vec)	0.029	0.861	0.904
BANN(BERT)	**0.021**	**0.886**	**0.935**

where n is the number of all test instances, k is the number of total aspect labels, y_i^j denotes the true value about j-th aspect of the i-th instances, $\hat{y}_i^j$ is the estimated value correspondingly, and $\mathbb{I}(.)$ is an indicator function which equals 1 if the condition in parentheses is true and 0 otherwise.
Accuracy measures how many instances have the right prediction of aspect list. In our experiment, only each element in the ground label set and predicted label set is identical, we regard this instance as the right one.
F1-measure is the harmonic mean between precision and recall. We employ the formula [15] and adapt it to multi-label tasks. The larger accuracy and F1-measure correspond to more outstanding performance.

$$F1 = \frac{2}{n}\sum_{i=1}^{n}\frac{|y_i \cap \hat{y}_i|}{|y_i|\,|\hat{y}_i|} \tag{8}$$

where y_i presents the ground truth label sets, and $\hat{y}_i$ presents the predicted label sets. For example, a given instance includes two aspects, which means its $|y_i|$ is 2.

- **Hyper-parameters** In our experiments, we split the corpus into training sets and test sets by the ratio of 4:1, In BERT word embeddings, vector dimension is fixed to 768. The max length of the question and the answer are 30, and the initial learning rate is 0.005. Moreover, all models are trained by mini-batch of 32 instances. Besides, all weight matrix and bias matrix are initialized by sampling from the uniform distribution $U(-0.01, 0.01)$, and the dropout rate is 0.5.

4.2 Experimental Results

To comprehensively verify the effectiveness and advantages of our proposed model, we design a series of models as baselines on the same corpus.
Binary Relevance: This approach transforms the multi-labeled aspect classification into multiple binary classifications without label correlation [16].
Classifier Chains: This approach addresses the multi-labeled aspect classification by a chain of binary classifies, which considers the label correlation [16].
LSTM(Word2vec): This approach leverages Word2vec to transform the inputs, and captures the contextual information via LSTM [3].
BANN(Word2vec): Our model employs a bidirectional attention layer instead of simple LSTM and uses Word2vec embedding.
BANN(BERT): Our final model employs a bidirectional attention layer and uses BERT embedding instead of Word2vec.

Table 3. Ablation studies results

Models	Hamming loss	Accuracy	F1-measure
BANN(BERT)	0.021	0.886	0.935
- LSTM	0.081	0.559	0.606
- Attention	0.029	0.849	0.911
- Q2A Attention	0.082	0.629	0.675
- A2Q Attention	0.031	0.855	0.911
Using weighted summation instead of averaging	**0.019**	**0.896**	**0.942**

We performed experiments of each approach above, and the results are shown in Table 2 with the mean value of 10 times experiments.

From Table 2, we can see that all deep neural network methods outperform traditional machine learning methods including **Binary Relevance** and **Classifier Chains** by a large margin, showing that deep neural network methods can learn more complicated aspect information in our QA-AC task.

In addition, our proposed **BANN(Word2vec)** has a better performance than **LSTM(Word2vec)** with the same embeddings and achieves a reduction of 0.1% Hamming loss and the improvement of 1.8 % (Accuracy) and 0.1 % (F1-measure), which is a popular deep neural network method in a large proportion of NLP tasks. It suggests that our bidirectional QA attention neural network is significantly effective, which highlights the contextual QA matching information.

In the end, we restructure our model by BERT instead of Word2vec and achieve the best scores in all evaluation metrics, markedly with 0.1% reduction in Hamming loss, 4.3% increase in Accuracy and 3.4% in F1-measure, which shows that BERT contextualized representation can bring performance improvement for our aspect classification task and proves the superiority of BERT. Significance test shows that the improvement of our model is significant ($p - value < 0.05$).

4.3 Ablation Studies

We conduct ablation studies on our BANN model and expose the results in Table 3 for evaluating the individual contribution of different components. The ablation of the LSTM layer results much more Hamming loss and a drop of over 30% on both Accuracy and F1-measure, which shows the consequence of characters' order information. The bidirectional attention accounts for about 3.7% of performance degradation on Accuracy and 2.4% on F1-measure, but the importance of these two directions are not balanced. By comparing the ablations of different directional attention, we can see that the effect of ablating Q2A attention is more severe than ablating A2Q attention. This is because the question contains more aspect-related information while the answer contains more sentiment polarities information in QA style reviews. Finally, we substitute the strategy of weighted summation for averaging in coalescing 12 transformer layers representations from BERT to get the contextualized word embeddings. The result manifests that the weighted summation improves performance by 1% on Accuracy and 0.7% on F1-measure while reduces Hamming loss by 0.2% via generating more rational word representations.

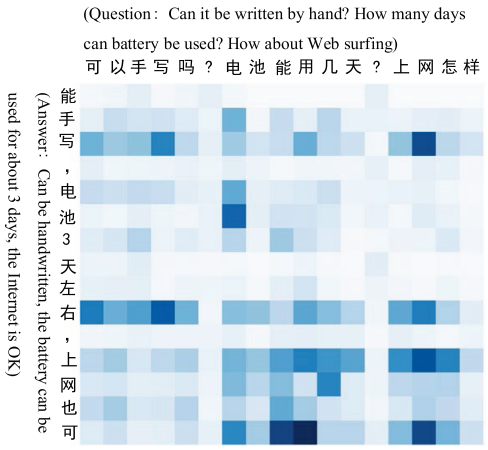

Fig. 4. An attention visualization with the aspects of "Function" and "Battery".

4.4 Visualized Analysis

Figure 4 depicts the A2Q attention weights from our BANN, which are used to measure the importance of characters in the question according to the matching information between question and answer. The sample review contains two aspects: "Function" and "Battery". It is apparent that there are six characters related to aspects in question including "手", "写", "电", "池", "上" and "网", which also appears in answer. By carefully inspecting the color depth on these characters, we can see that our BANN emphasizes the importance and relationship between the similar aspect-related characters in question and answer. After bidirectional attention layer, our model can enhance the weights of aspect-related characters in "手写", "电池" and "上网", and decrease the value of characters in "3 天", "怎样" and punctuation characters.

5 Conclusion

In this paper, we propose a contextualized attention-based model for aspect classification on QA style reviews. Firstly we recast the QA-AC task as a multi-label classification problem. Then we capture the contextual information by BERT model, where we employ two strategies to aggregate multiple transformer layers outputs. Further, we use the BANN model to incorporate the QA matching information between questions and answers. Experimental results demonstrate that our approach outperforms several widely-used baselines significantly. For future work, we will explore the effectiveness of joint learning on QA-AC task by jointing aspect classification and aspect-level sentiment classification.

Acknowledgements

This work is supported in part by Industrial Prospective Project of Jiangsu Technology Department under Grant No.BE2017081 and the National Natural Science Foundation of China under Grant No.61572129.

References

1. Fan, F., Feng, Y., Zhao, D.: Multi-grained attention network for aspect-level sentiment classification. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 3433–3442 (2018)
2. He, R., Lee, W.S., Ng, H.T., Dahlmeier, D.: Exploiting document knowledge for aspect-level sentiment classification. arXiv preprint arXiv:1806.04346 (2018)
3. Wang, Y., Huang, M., Zhao, L., et al.: Attention-based lstm for aspect-level sentiment classification. In: Proceedings of the 2016 conference on empirical methods in natural language processing. pp. 606–615 (2016)
4. Rubtsova, Y., Koshelnikov, S.: Aspect extraction from reviews using conditional random fields. In: International Conference on Knowledge Engineering and the Semantic Web. pp. 158–167. Springer (2015)
5. Xu, H., Liu, B., Shu, L., Yu, P.S.: Double embeddings and cnn-based sequence labeling for aspect extraction. arXiv preprint arXiv:1805.04601 (2018)
6. He, R., Lee, W.S., Ng, H.T., Dahlmeier, D.: An unsupervised neural attention model for aspect extraction. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 388–397 (2017)
7. Wu, H., Liu, M., Wang, J., Xie, J., Shen, C.: Question-answering aspect classification with hierarchical attention network. In: Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data, pp. 225–237. Springer (2018)
8. Dai, A.M., Le, Q.V.: Semi-supervised sequence learning. In: Advances in neural information processing systems. pp. 3079–3087 (2015)
9. Kim, Y., Jernite, Y., Sontag, D., Rush, A.M.: Character-aware neural language models. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)
10. Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. arXiv preprint arXiv:1802.05365 (2018)
11. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training. URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf (2018)
12. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
13. Adhikari, A., Ram, A., Tang, R., Lin, J.: Docbert: Bert for document classification. arXiv preprint arXiv:1904.08398 (2019)
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
15. Godbole, S., Sarawagi, S.: Discriminative methods for multi-labeled classification. In: Pacific-Asia conference on knowledge discovery and data mining. pp. 22–30. Springer (2004)
16. Szymański, P., Kajdanowicz, T.: A scikit-based Python environment for performing multi-label classification. ArXiv e-prints (Feb 2017)