Aspect-based Sentiment Analysis in Question Answering Forums *

Wenxuan Zhang[1], Yang Deng[1], Xin Li[2], Lidong Bing[2] and Wai Lam[1]
[1]The Chinese University of Hong Kong
[2]DAMO Academy, Alibaba Group
{wxzhang,ydeng,wlam}@se.cuhk.edu.hk
{xinting.lx,l.bing}@alibaba-inc.com

Abstract

Aspect-based sentiment analysis (ABSA) typically focuses on extracting aspects and predicting their sentiments on individual sentences such as customer reviews. Recently, another kind of opinion sharing platform, namely question answering (QA) forum, has received increasing popularity, which accumulates a large number of user opinions towards various aspects. This motivates us to investigate the task of ABSA on QA forums (ABSA-QA), aiming to jointly detect the discussed aspects and their sentiment polarities for a given QA pair. Unlike review sentences, a QA pair is composed of two parallel sentences, which requires interaction modeling to align the aspect mentioned in the question and the associated opinion clues in the answer. To this end, we propose a model with a specific design of cross-sentence aspect-opinion interaction modeling to address this task. The proposed method is evaluated on three real-world datasets and the results show that our model outperforms several strong baselines adopted from related state-of-the-art models.

1 Introduction

Aspect-based sentiment analysis (ABSA) usually involves two sub-tasks including aspect term extraction (ATE) and aspect sentiment classification (ASC) (Liu, 2012; Pontiki et al., 2014). For an example sentence "*The feel of the restaurant was crowded but the food is great.*", ATE is to detect the mentioned aspects "*feel*" and "*food*", whereas supposing aspects are given, ASC predicts their sentiment polarities as negative and positive respectively. Given the broad application scenarios, the two sub-tasks (He et al., 2017; Sun et al., 2019; Tulkens and van Cranenburgh, 2020) and their joint prediction (Li et al., 2019a; Chen and Qian, 2020a; Mao et al., 2021; Zhang et al., 2021) have received increasing attention in recent years.

Q: How about the *screen*? Is this phone's *battery life* durable? Thanks in advance!
A: Not as large as I thought. But the battery is quite good, I like it.

TASK	INPUT	OUTPUT
ATE-QA	QA pair	[screen]; [battery life]
ASC-QA	QA pair + [screen]	NEG
	QA pair + [battery life]	POS
ABSA-QA	QA pair	[screen]$_{\text{NEG}}$ [battery life]$_{\text{POS}}$

Figure 1: Demonstrations of ABSA-QA task and its two sub-tasks including ATE-QA and ASC-QA.

Most existing ABSA studies focus on a single opinionated sentence such as the customer review (Pontiki et al., 2014, 2015). Besides product reviews, another kind of opinion sharing platform, namely question answering (QA) forum, has been provided on many E-commerce websites, due to the rising demand for users and sellers to communicate with the former buyers to obtain their opinions towards various aspects of the concerned product (Zhang et al., 2020b). Thus, investigating the ABSA task on such QA forums (denoted as **ABSA-QA**) can be a meaningful problem for revealing the rich opinion information from those QA pairs.

Several attempts have been made on analyzing the sentiment information in QA forums. However, they either predict an overall sentiment polarity towards the entire QA pair (Shen et al., 2018; Hu et al., 2020) or only consider partial ABSA-QA problems. For example, Wang et al. (2019) tackle the ASC-QA task under the assumption that the targeted aspects are given. As illustrated in Figure 1, they perform aspect-level sentiment classification according to both of the QA pair and the input aspect. However, obtaining the discussed aspects is not a trivial task, which is especially difficult

* The work described in this paper is substantially supported by a grant from the Research Grant Council of the Hong Kong Special Administrative Region, China (Project Code: 14200719). Work partially done when Wenxuan Zhang was an intern at Alibaba.

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 4582–4591
November 7–11, 2021.

for those QA pairs involving multiple aspects. Inspired by previous success on jointly solving the two sub-tasks in review-oriented ABSA (He et al., 2019; Luo et al., 2019; Chen and Qian, 2020a), we aim to handle the ABSA-QA task in a similar unified setting in this work[1]. As shown in Figure 1, given a question-answer pair, our goal is to jointly detect the discussed aspect(s) and predict their corresponding sentiment polarities.

To tackle the ABSA-QA task, an intuitive idea would be concatenating the question and answer sentence, then employing the existing ABSA models to solve it. However, the question and answer sentence are two parallel sequences, therefore, simply concatenating them cannot produce a semantic-fluent expression. In such a concatenation, the aspect terms and their corresponding opinion words do not appear next or near to each other, making the position clue utilized by many ABSA models, i.e., the aspect modifier is closer to the corresponding aspect term in the sentence, invalid (Hu et al., 2019; He et al., 2019). To make matters worse, it will result in wrong proximity relation, for instance, compared with "*quite good*", "*not as large as*" is nearer to "*battery life*" in the example. Meanwhile, because the opinions are expressed in an interactive manner, i.e., the question asks about one or multiple aspects and the answer expresses the opinions towards them, the aspect terms are likely to be omitted or rephrased in the answer sentence. Returning to the example in Figure 1, the aspect "*battery life*" is shortened to "*battery*" while the explicit mention of the aspect "*screen*" is directly omitted in the answer. This requires the model to capture the aspect-opinion interactions between the QA pair to align the concerned aspect in the question with the opinions expressed in the answer instead of simply treating them as a consecutive sequence.

In this paper, we formulate the ABSA-QA task as a sequence labeling problem on the question text with the unified tagging scheme denoting both the aspect boundary and sentiment polarity for each word. Because of the interactive nature of QA pairs, when predicting the unified tags for the question text, it is essential to utilize the answer information to locate the aspect terms as well as predict their sentiments. To this end, we propose a novel model with cross-sentence aspect-opinion interaction modeling to tackle the ABSA-QA task. Specifically, our model is built on top of the pre-trained BERT network, which has shown its effectiveness in the general ABSA problem (Hu et al., 2019; Li et al., 2019b; Chen and Qian, 2020a). Firstly, to capture the interactions between the question and answer, an inter-QA attention mechanism is employed, which aligns the aspect in the question with the corresponding opinions in the answer. A gated fusion layer is then designed for combining the information from the answer and the question itself to obtain an enriched aspect-aware question representation. Next, we employ attentive encoding to summarize the main opinion information from the answer text into the question representation and use two types of CNN layers to refine the final representation and control the sentiment consistency. Finally, the refined feature representation for each question token is fed to a linear layer to predict the unified tag. In addition to the base model described above, we exploit two auxiliary tasks to further enhance it: (i) An auxiliary aspect term extraction task is introduced to better guide the learning of the aspect-aware question representation. (ii) To improve the interaction modeling across the sentence pair, we propose to pre-train the related components with QA pair matching task for obtaining the prior knowledge on aligning two sentences.

In summary, our main contributions are as follows: (1) We study the ABSA-QA task, aiming to jointly detect the discussed aspects and their sentiment polarities for a given QA pair. (2) We propose a model that carefully captures the cross-sentence aspect-opinion interactions and utilize two auxiliary tasks for better feature representation learning to tackle the concerned task. (3) We conduct extensive experiments on real-world datasets across three domains and the results show that our proposed model outperforms several strong baselines adopted from related state-of-the-art models.

2 Methodology

We formulate the ABSA-QA task as a sequence labeling problem on the question text and employ a unified tagging scheme: $\mathcal{Y}^u = \{\texttt{B}, \texttt{I}, \texttt{E}, \texttt{S}\}$-$\{\texttt{POS}, \texttt{NEU}, \texttt{NEG}\} \cup \{\texttt{O}\}$ to jointly denote the aspect term and its sentiment polarity for each token following (Li et al., 2019a). The former part of the tag defines the boundary of the aspect whereas the latter refers to its sentiment polarity. Given a QA pair including a question $Q = \{q_1, q_2, \ldots, q_m\}$ and its

[1]Since some early studies use "ABSA" to refer to ASC task, recent works use "unified/end-to-end ABSA" to emphasize the joint solution of two sub-tasks. Following this convention, we use "(unified) ABSA-QA" to refer to our task in this paper.

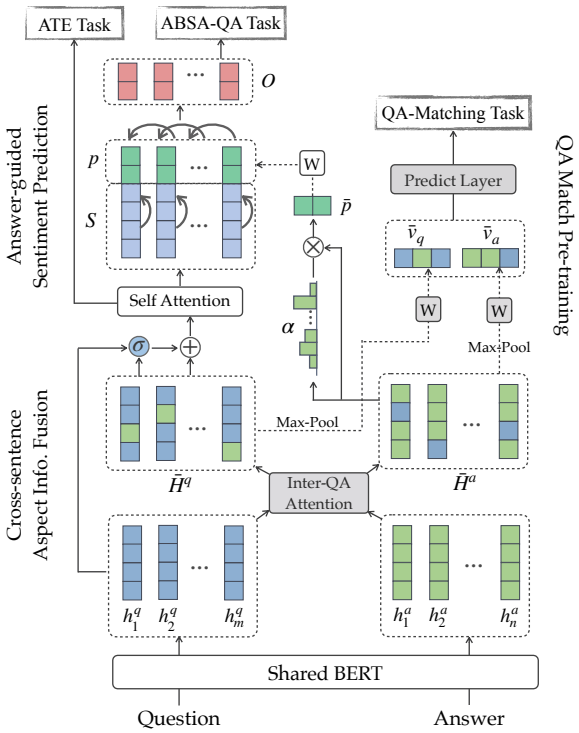

Figure 2: Architecture of our proposed model

answer $A = \{a_1, a_2, \ldots, a_n\}$, we aim to detect the discussed aspects and their sentiment polarities by predicting a tag sequence $Y = \{y_1, y_2, \ldots, y_m\}$ for the question text where $y_i \in \mathcal{Y}^u$.

2.1 Model Overview

The overall architecture of our proposed model is depicted in Figure 2, which mainly consists of three components, including cross-sentence aspect information fusion, answer-guided sentiment prediction, and QA matching pre-training. Given a QA pair, we first utilize inter-sentence attention to capture the interactions between the question and answer sentences for aligning the aspects with their corresponding opinion information. A gated fusion layer and a self-attention layer are then employed to fuse and refine the feature representation. To summarize the expressed opinion, we then conduct a self-attentive encoding on the answer for highlighting the sentiment. A local context encoder is then applied to maintain the sentiment consistency. Finally, the refined question representation is utilized to predict the tag sequence with the unified tagging scheme. To learn a better aspect-aware question representation, our model is jointly trained with an auxiliary aspect term extraction (ATE) task, which makes use of the attended information from the answer to help extract the discussed aspect.

In addition, an auxiliary QA matching task aiming at measuring the relevance between a QA pair is conducted. As shown in the shaded modules in Figure 2, it utilizes the interacted representations of two sentences to make the prediction, thus its inter-QA attention module can be equipped with the knowledge of capturing the alignment of related elements between the QA pair. We adopt the pre-training strategy where the trained parameters from the QA matching task are used to initialize the certain network modules of the main model.

2.2 Model Description

We use BERT (Devlin et al., 2019) as our backbone network to obtain contextualized word representations. Given a question Q and an answer A, we employ BERT to transform each token w_i to its word vector $h_i \in \mathbb{R}^{d_h}$ where d_h is the hidden dimension. We denote the transformed sequences as $H^q = \{h_1^q, h_2^q, \ldots, h_m^q\}$ and $H^a = \{h_1^a, h_2^a, \ldots, h_n^a\}$ respectively, where m and n are their sequence lengths. Following this notation convention, we shall use capital letter such as H^q to denote the matrix of the whole sequence and the corresponding lowercase letter such as h_i^q to refer to the representation of the i-th token hereafter.

2.2.1 Cross-sentence Aspect Information Fusion

To align the mentioned aspects with their opinion information and capture the complete semantic information of the QA pair, it requires to model the interactions across the question and answer sentences. To this end, we employ an inter-sentence attention mechanism to conduct the matching between them. Specifically, we define an attention operation $\textsc{Attn}(X, Y)$ between the sequence X and Y as follows:

$$\textsc{Attn}(X, Y) = \textsc{Ln}(X + \textsc{Mh-Att}(X, Y, Y)) \quad (1)$$

where $\textsc{Mh-Att}(Q, K, V)$ is the multi-head attention operation described in (Vaswani et al., 2017), and $\textsc{Ln}$ denotes the layer normalization (Ba et al., 2016). Then we can compute an answer-attended question representation as $\bar{H}^q = \textsc{Attn}(H^q, H^a)$. From the perspective of the multi-head attention mechanism, such representations can be regarded as the results of using the question as the "query" to align with the "key" in the answer so as to obtain the related opinion information, which is the "value" part. Similarly, we can also obtain

the question-attended answer representation as $\bar{H}^a = \textsc{Attn}(H^a, H^q)$.

The matched information from the answer can well indicate the mentioned aspects. For example, it may rephrase or simply repeat the aspect term asked in the question and then present their sentiment. To combine the attended representations $\bar{H}^q$ and the original representations H^q, a multi-layer perceptron is typically involved in solving the text matching task (Chen et al., 2017; Yang et al., 2019). However, since we are tackling a token-level prediction problem, such a fusion method would obscure the fine-grained feature representations. We propose a gated fusion approach to absorb the aspect information from the answer while also maintain the most salient information in each question token. Concretely, for the i-th word, we have:

$$g = \sigma(W^r h_i^q + W^a \bar{h}_i^q + b_g) \quad (2)$$

$$\tilde{h}_i^q = g \odot h_i^q + (1 - g) \odot \bar{h}_i^q \quad (3)$$

where W^r and W^a are trainable parameters, σ and $\odot$ denote the sigmoid function and the element-wise multiplication respectively. The resulting $\tilde{h}_t^q$ represents the fused representation for the i-th question token. The attention operation is then applied on top of it to further refine the representation after the fusion as $S = \textsc{Attn}(\tilde{H}, \tilde{H})$, which is in essence the self-attention module in the transformer network (Vaswani et al., 2017).

2.2.2 Answer-guided Sentiment Prediction

To more explicitly highlight the sentiment polarity expressed in the answer sentence, we next conduct self-attentive encoding on the answer text to emphasize the most important part in it:

$$\alpha_i = \frac{\exp(w_s^T \tanh(W^s \bar{h}_i^a))}{\sum_{k=1}^{n} \exp(w_s^T \tanh(W^s \bar{h}_k^a))} \quad (4)$$

where $w_s \in \mathbb{R}^{d_a}$ and $W^s \in \mathbb{R}^{d_a \times d_h}$ are trainable parameters, α_i denotes the weight for the i-th answer token. We then compute a fixed-size answer representation $\bar{p}$ as follows:

$$\bar{p} = \sum_{i=1}^{n} \alpha_i \bar{h}_i^a \quad (5)$$

which summarizes the main opinion information in the answer. A linear transformation is further applied to obtain a more condensed representation $p \in \mathbb{R}^{d_e}$. We concatenate it to each question token to enlarge the sentiment information and denote the new question representation as $\bar{S}$ where $\bar{s}_i = [s_i; p]$, and $[;]$ is the concatenation operation.

Given the concatenated representations, we first adopt a point-wise CNN network to refine the feature for each question token, where the kernel size is set to one for only considering each token itself. Then another CNN layer with larger kernel size is stacked on top of it to exploit the neighboring information for each token, which helps control the sentiment consistency to avoid different sentiments are predicted for the same aspect:

$$O = \mathrm{ReLU}(W^l * \mathrm{ReLU}(W^t * \bar{S} + b_t) + b_l) \quad (6)$$

where W^l and W^t denote the trainable parameters of two convolutional kernels, $*$ refers to the convolution operation. $O \in \mathbb{R}^{m \times d_u}$ is the final feature representation for the entire question sequence.

2.2.3 Model Training

After obtaining the final representation o_i for each question token, the probability score $\hat{y}_i$ over the unified tagging set $\mathcal{Y}^u$ can be computed through a linear layer. The cross-entropy loss $\mathcal{L}^{\mathcal{U}}$ for the main ABSA-QA task is then calculated as follows:

$$\hat{y}_i = \mathrm{Softmax}(W^u o_i + b_u) \quad (7)$$

$$\mathcal{L}^{\mathcal{U}} = -\sum_{i=1}^{m} y_i \log(\hat{y}_i) \quad (8)$$

In the model described above, the ABSA-QA task is tackled with two main steps where we first focus on the aspect-level information, then predict the sentiment polarity, both with interacted answer information. To enforce better aspect-aware question representation, we incorporate the ATE task at the connection of these two phases. Concretely, the question representation S is used to predict a tag sequence denoting the boundary of the aspect:

$$\hat{z}_i = \mathrm{Softmax}(W^z s_i + b_z) \quad (9)$$

$$\mathcal{L}^{\mathcal{T}} = -\sum_{i=1}^{m} z_i \log(\hat{z}_i) \quad (10)$$

where W^z is a weight matrix, $\hat{z}_i$ is the predicted score of the i-th question token over the boundary tag set $\mathcal{Y}^z = \{\mathtt{B}, \mathtt{I}, \mathtt{O}, \mathtt{E}, \mathtt{S}\}$, z_i is the ground-truth label, $\mathcal{L}^{\mathcal{T}}$ is the cross-entropy loss for the ATE task. Note that although the target of the ATE task is already contained in the main ABSA-QA task, augmenting the ATE task here can more explicitly guide the learning of aspect-aware question representation, helping the following components for solving the entire task.

To train the overall framework, the loss of the main ABSA-QA task and the auxiliary ATE task are combined in a multi-task learning paradigm to form the final loss $\mathcal{L}$:

$$\mathcal{L} = \mathcal{L}^{\mathcal{U}} + \lambda \mathcal{L}^{\mathcal{T}} \tag{11}$$

where λ is a hyper-parameter to control the influence of the ATE task.

2.3 Pre-training with QA Matching Task

To better capture the cross-sentence interaction, we propose to pre-train the corresponding network modules with an auxiliary QA matching task for transferring some prior knowledge of aligning related elements between QA pairs. Specifically, after obtaining the attended representations $\bar{H}^q$ and $\bar{H}^a$, we conduct a max-pooling on them to obtain the vector representations v_q and v_a:

$$v_q = \text{Max-Pool}(\bar{H}^q);\ v_a = \text{Max-Pool}(\bar{H}^a) \tag{12}$$

which are then transformed with a linear layer to obtain fixed-size vector representations containing the main semantic information, denoted as $\bar{v}_q$ and $\bar{v}_a$ respectively. The prediction layer then takes the two encoded representations to predict the matching between the QA pair following (Chen et al., 2017; Yang et al., 2019):

$$\hat{x} = \text{G}([\bar{v}_q; \bar{v}_a; \bar{v}_q - \bar{v}_a; \bar{v}_q \odot \bar{v}_a]) \tag{13}$$

where $\text{G}(\cdot)$ is a multi-layer perceptron, $\hat{x}$ is the predicted score, which can be used to calculate a cross-entropy loss with the ground-truth label to train this matching task end-to-end. Note that our main target of conducting such matching task is to equip the interaction layer with better alignment capabilities, so we keep the design of the network architecture here in a simple manner.

Since the original QA data are already paired, i.e., the matching labels between them are always true. For each question, we randomly sample an answer of other questions in the training data to construct a "negative" QA pair. The augmented training data is then used to pre-train the interaction layer, and the trained weights are utilized as the initialization of the corresponding network parameters of the main ABSA-QA model.

Dataset		Train	Test	Total
ELEC	# QA pair	3639	909	4548
	# aspect	4071	1018	5089
BEAUTY	# QA pair	3577	894	4471
	# aspect	3887	964	4851
BAGS	# QA pair	3620	904	4524
	# aspect	4228	1035	5263

Table 1: Statistics of the datasets of three domains.

3 Experiments

3.1 Datasets

We conduct experiments with QA pairs originally collected by Wang et al. (2019) from *Taobao*[2], the biggest E-commerce platform in China. It includes datasets from three product categories, namely Electronics (ELEC), Beauty (BEAUTY) and Bags (BAGS). Each QA pair is annotated with one or multiple tuples: (aspect term, polarity) where the aspect term is a span of the question text. We remove the duplicated QA pairs in the original corpus and filter out the mis-annotated data[3]. For each product category, we randomly split the data into training and testing set with the ratio of 8:2. During the training phase, we randomly sample 20% of the training data as the development data to tune the hyper-parameters and use the rest for training. The detailed statistics of each dataset including the number of QA pairs and aspect terms are summarized in Table 1.

The model achieving the best performance on the development set is used for evaluation on the test set. We adopt the F1 score as the main evaluation metric and also report the corresponding precision (Pre) and recall (Rec) scores. The measurement are based on exact match where a prediction is correct only when the extracted span and the predicted sentiment are both correct. Average scores over 5 runs with different random initialization are reported.

3.2 Comparison Methods

We compare with the following methods:

BiLSTM-CRF: a baseline model with Bidirectional LSTM network as the encoding module and a CRF layer as the label decoding module. The unified tagging scheme is adopted.

E2E-TBSA (Li et al., 2019a): an end-to-end model for tackling ATE task and ASC task simultaneously

[2] https://www.taobao.com/

[3] There are some QA pairs whose aspect terms do not appear in the question nor the answer text due to misspelling.

Model	ELEC			BEAUTY			BAGS		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
BiLSTM-CRF	77.54	70.40	73.73	74.24	65.87	69.78	81.46	73.86	77.47
E2E-TBSA	84.36	77.30	80.67	75.58	71.92	73.71	84.85	80.96	82.86
BERT-Linear	81.29	85.79	83.47	75.11	80.44	77.67	82.14	88.48	85.18
BERT-GRU	81.71	86.48	84.02	78.31	81.78	78.41	83.42	88.08	85.68
BERT-SAN	82.79	86.76	84.72	75.54	81.19	78.25	83.81	88.44	86.06
Span-Joint	85.93	85.87	85.89	81.21	79.78	80.48	87.14	86.04	86.57
Span-Pipeline	84.65	89.51	87.01	79.89	81.92	80.89	85.31	89.71	87.41
BERT-QA	84.41	88.19	86.25	79.41	82.77	81.05	85.88	89.18	87.49
Base Model	85.99	87.87	86.92	80.70	83.31	81.99	87.83	90.35	89.07
Base+ATE	86.77	88.05	87.39	82.19	83.08	<u>82.63</u>	87.65	90.69	89.13
Base+QA	87.11	88.66	<u>87.87</u>	81.92	83.31	82.60	87.91	90.91	**89.38**
Full Model	88.39	88.48	**88.44**	82.88	82.86	**82.87**	87.71	90.86	<u>89.26</u>

Table 2: Main results of the ABSA-QA task. The best performance are in bold and the second best performance are underlined.

with the unified tagging scheme. We use the officially released code[4] to obtain the results.

Bert-Linear (Devlin et al., 2019): the original BERT model with a single linear layer stacked on top of the last transformer block to conduct the question tagging task.

BERT-{GRU, SAN} (Li et al., 2019b): two BERT-based models with specific ABSA layers achieving the best performance on its investigated two datasets respectively. Bert-GRU uses Gated Recurrent Unit (GRU) with additional layer normalization as the ABSA layer, while Bert-SAN model uses a single layer self-attention network.

Span-Joint (Hu et al., 2019): a span-based model for jointly performing ATE and ASC tasks with BERT as the backbone. The "Span-Joint" variant has two output layers on top of the same encoder, one for each task. We run the released code[5] to produce the results.

Span-Pipeline (Hu et al., 2019): a state-of-the-art method for the unified ABSA task. It includes a multi-target extractor and a polarity classifier, both with BERT as the base network. Two models are separately trained and piped together to make predictions during inference.

BERT-QA (Sun et al., 2019): It transforms the ABSA task to a sentence pair classification task[6]. We adopt "BERT-pair-QA-M" variant and change its output layer to conduct token-level classification. It serves as a strong baseline for our concerned ABSA-QA task.

For those models using the same unified tagging scheme as ours, we concatenate the question and answer sequences as their inputs for them to utilize the answer information.

For our proposed model, we report the results for the following variants: **Base Model**, which only uses $\mathcal{L}^{\mathcal{U}}$ to train the model; **Base+ATE**, where the base model is augmented with the ATE task using $\mathcal{L}$ as the loss function; **Base+QA**, where the base model is augmented with the pre-training of QA pair matching; **Full Model**, our full model involving both auxiliary tasks.[7]

3.3 Experimental Settings

For baseline models using pre-trained word vectors, we use `cc.zh.300.vec`[8] trained with fastText (Bojanowski et al., 2017) for fair comparison. For BERT-based models including ours, we use the same pre-trained `BERT-Base, Chinese`[9] in all experiments, which includes 12 transformer layers and the hidden dimension d_h is 768. For our proposed model, the parameters of BERT is further fine-tuned during the training process.

Regarding the network architectures, the hidden dimension of the answer encoding module d_a is 300, the dimension of the encoded answer vector d_e is 64. For the local context capturing layer, the

[4] https://github.com/lixin4ever/E2E-TBSA

[5] https://github.com/huminghao16/SpanABSA

[6] https://github.com/HSLCY/ABSA-BERT-pair

[7] The code is publicly available at https://github.com/IsakZhang/ABSA-QA.

[8] https://github.com/facebookresearch/fastText/blob/master/docs/crawl-vectors.md

[9] https://github.com/google-research/bert

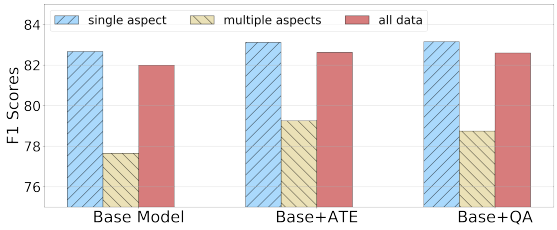

Figure 3: Performance on the BEAUTY dataset of QA pairs involving single and multiple aspects respectively.

kernel sizes are 1 and 3 respectively and the dimension of the encoded question representation is 256. λ is set to 0.5 in Eq. 11. We also conduct dropout after the BERT encoding layer and before the output layer (in Eq. 7) with dropout rate both being 0.1. Our model is trained using Adam optimizer with the learning rate being 3e-5. The batch size is set to 25 for all datasets. The experiments are conducted on a single GeForce GTX 1080 Ti GPU.

3.4 Results and Analysis

3.4.1 Main Results

Table 2 reports the performance of our proposed model and baseline methods on the concerned ABSA-QA task. We can see that our model consistently outperforms those strong baselines adopted from state-of-the-art models and gives the best F1 score across all datasets.

Among the baseline methods, it can be observed that the BERT-QA model provides a strong baseline performance. Compared with Span-Pipeline which is a state-of-the-art ABSA model for single-sentence, BERT-QA still obtains better performance in 2 out of 3 datasets, showing the importance of explicitly considering the input data format (i.e., QA pair) rather than simply treating it as a consecutive sequence for the ABSA-QA task. Our proposed model, even the base variant, outperforms the strongest baseline in all domains, suggesting a carefully-designed cross-sentence interaction modeling is beneficial on the concerned task. Another finding is that BERT-based methods, even the simplest BERT-Linear outperforms E2E-TBSA, which is a state-of-the-art non-BERT model, demonstrating the superior power of BERT for capturing the contextual information of the input sentence.

3.4.2 Impact of Two Auxiliary Tasks

Comparing the different variants of our proposed method, assisting the base model with the ATE task achieves better performance in all domains. This result indicates that ABSA-QA can benefit from jointly learning with aspect term extraction task, which enables the model to explicitly learn a better aspect-aware question representation. Utilizing the QA pair matching task to pre-train the interaction layer also brings in some performance gain, which shows that such pre-training strategy effectively enhance the inter-sentence attention layer with better capabilities to align the aspect-opinion information across two parallel sentences. To further investigate such improvements, we report the F1 scores on the BEAUTY dataset for QA pairs containing single and multiple aspects respectively in Figure 3. We can see that there is a significant performance boosting on those difficult data instances with multiple aspects when incorporating the ATE task or QA pre-training. However, as shown in Table 2, utilizing both tasks does not necessarily lead to the best performance, e.g., the results on the BAGS dataset. This is likely due to the reason that the base model itself can already achieve good results (around 1.6% absolute gain compared with BERT-QA), while the auxiliary tasks make relatively slight contribution to the final performance.

	ELEC	BEAUTY	BAGS
Base Model+ATE	87.39	82.63	89.13
- w/o Q self attention	87.15	82.44	88.85
- w/o answer encoding	86.81	81.81	88.58
- w/o local context layer	87.10	82.38	88.43

Table 3: Ablation Study on Base Model+ATE

3.4.3 Ablation Study

To investigate the effectiveness of some important components of our proposed model, we conduct ablation studies on the "Base Model+ATE" variant and report F1 scores across three datasets in Table 3. As observed from the results, the model without the question self-attention ("w/o Q self attention") and without the final local context capturing layer ("w/o local context layer") both suffer from a performance decrease, showing the effectiveness of refining the feature representations after the question-answer interactions. Removing the answer sentiment encoding component ("w/o answer encoding"), i.e., using S instead of $\bar{S}$ in the Eq. 6 leads to a large performance fall. This result indicates that it is effective and necessary to integrate the opinion information in the answer into the question representation for a precise sentiment classification.

Examples	Span-Pipeline	Ours-Base	Ours-Full
$\mathbf{Q_1}$: [遮痘]$_{\text{NEG}}$怎么样？ How about [cover acne]$_{\text{NEG}}$? $\mathbf{A_1}$: 痘印能遮，痘痘遮不了。 It can cover the acne scar, cannot cover the acne.	[遮痘]$_{\text{POS}}$ ✗ [cover acne]$_{\text{POS}}$	[遮痘]$_{\text{NEG}}$ ✓ [cover acne]$_{\text{NEG}}$	[遮痘]$_{\text{NEG}}$ ✓ [cover acne]$_{\text{NEG}}$
$\mathbf{Q_2}$: 遮瑕哪样，[持久]$_{\text{NEG}}$不？？ How about mask blemishes? Can the effect [last long]$_{\text{NEG}}$? $\mathbf{A_2}$: 不持久 Didn't last long.	[遮瑕]$_{\text{POS}}$ ✗ [mask blemishes]$_{\text{POS}}$ [持久]$_{\text{NEG}}$ [last long]$_{\text{NEG}}$ ✓	[持久]$_{\text{NEG}}$ ✓ [last long]$_{\text{NEG}}$	[持久]$_{\text{NEG}}$ ✓ [last long]$_{\text{NEG}}$
$\mathbf{Q_3}$: 书包的[容量]$_{\text{POS}}$和[质量]$_{\text{POS}}$怎么样 How's the [capacity]$_{\text{POS}}$ and the [quality]$_{\text{POS}}$ of this backpack? $\mathbf{A_3}$: 都还可以吧，容量我是放假回家背的微电脑和5，6件衣服的样子 Both are okay. For the capacity, I bring a laptop and 5 or 6 clothes with me when I go home on holiday.	None ✗ [质量]$_{\text{POS}}$ ✓ [quality]$_{\text{POS}}$	[容量]$_{\text{POS}}$ ✓ [capacity]$_{\text{POS}}$ [质量]$_{\text{POS}}$ ✓ [quality]$_{\text{POS}}$	[容量]$_{\text{POS}}$ ✓ [capacity]$_{\text{POS}}$ [质量]$_{\text{POS}}$ ✓ [quality]$_{\text{POS}}$
$\mathbf{Q_4}$: 你们的手机[质量]$_{\text{NEG}}$怎么样？我手机弯曲了。 How's the [quality]$_{\text{NEG}}$ of your phones, mine is already bent. $\mathbf{A_4}$: 触屏经常没反应，数据流量很慢，先说明我不是在偏僻的地方。Touching screen often does not react. The network flow is very slow, just be clear that I'm not in a remote area.	[质量]$_{\text{POS}}$ ✗ [quality]$_{\text{POS}}$	[质量]$_{\text{POS}}$ ✗ [quality]$_{\text{POS}}$	[质量]$_{\text{NEG}}$ ✓ [quality]$_{\text{NEG}}$

Table 4: Case analysis. The "Examples" column contains sample QA pairs with gold labels where words in brackets are annotated aspect terms, the subscripts denotes their sentiment polarities. "None" in predictions denotes that no aspect terms are extracted. The correct/incorrect predictions are marked with ✓/✗ respectively.

3.4.4 Case Analysis

We present some sample cases including input QA pairs and predictions given by the baseline Span-Pipeline model, our proposed base model and the full model in Table 4. We can see that Span-Pipeline fails when the alignment is needed between the question and answer sentences. For example, the second answer $\mathbf{A_2}$ only comments on the "last long" aspect, thus Span-Pipeline just randomly assigns a sentiment polarity for the "mask blemishes". Regarding the third question $\mathbf{Q_3}$, its answer expresses "okay" to both aspects mentioned in the question, but only "quality" is detected by Span-Pipeline. Our proposed model, both the base and full model successfully handle these two cases, showing the necessity to model the interactions between the given QA pairs. For the last example $\mathbf{Q_4}$, the answer does not provide any direct comment on the asked aspects, for instance, it does not mention aspect "quality" or any related opinion term such as "bad" at all, making it difficult to predict the sentiment polarity. Our proposed full model equipped with the QA matching pre-training gives correct predictions on them, which attributes to the pre-training that brings in some prior knowledge for identifying that the answer is talking about the "quality" of the product.

4 Related Work

Aspect-based sentiment analysis (ABSA) has been extensively studied in recent years (Liu, 2012; Zhang et al., 2018). It is often decomposed into two sub-tasks. The first aspect term extraction (ATE) task aims to detect the mentioned aspect (He et al., 2017; Xu et al., 2019; Tulkens and van Cranenburgh, 2020; Li et al., 2020; Wei et al., 2020). The second aspect sentiment classification (ASC) task then predicts the sentiment polarity, assuming an aspect is given (Sun et al., 2019; Tang et al., 2020; Chen et al., 2020b; Zheng et al., 2020).

Since separately handling these two tasks ignores the relations between them and leads to unsatisfactory performance, recent works attempt to solve it in a unified framework. These studies either adopt a unified tagging scheme (Li et al., 2019b,a; Hu et al., 2019) or solving them in a multi-task learning paradigm with shared feature representations (He et al., 2019; Luo et al., 2019). Recently, there are also some attempts of combining another related task, namely opinion term extraction (OTE), with the ATE and/or ASC tasks to provide a more complete understanding of the aspect-level user sentiment (Chen et al., 2020a; Zhao et al., 2020; Chen and Qian, 2020b; Liang et al., 2020; Zhang et al., 2021).

However, most existing studies target at customer reviews (Pontiki et al., 2014, 2015) or twitter posts (Mitchell et al., 2013). Thus the proposed methods are often tailored for observations made in single-sentence situation. For example, many models consider the position clues between the aspect term and the opinion terms since they often appear next or near to each other in the reviews (Hu et al., 2019; He et al., 2019). Given the rising popularity of question answering (QA) forums (Zhang et al., 2020b,a; Deng et al., 2020), some studies aim at extracting sentiment information on them. Shen

et al. (2018) treat the QA pair as a whole and predict its sentiment polarity. Hu et al. (2020) further consider the syntax information in QA to improve the prediction. However, these work ignore the aspect-level information and only predict "conflict" if there are multiple aspects involved. Wang et al. (2019) focus on the ASC task in the QA forums (ASC-QA) which assumes the aspect is already given for the classification. Unlike these existing work, we investigate the unified ABSA-QA task in this paper, aiming to jointly tackle the ATE-QA and ASC-QA problem.

5 Conclusions

In this paper, we investigate the aspect-based sentiment analysis in question answering forums (ABSA-QA), aiming to jointly detect the discussed aspects and their sentiment polarities for a given QA pair. We demonstrate the challenges of conducting ABSA in QA settings and propose a model with carefully designed cross-sentence aspect-opinion interaction to tackle the task. Moreover, we utilize two auxiliary tasks including aspect term extraction task for learning better aspect-aware representation and QA pair matching task to pre-train the inter-QA attention components to for better aligning the question and answer sentence. Extensive experiments are conducted on three real-world datasets, showing the superiority of our proposed model against various baselines.

References

Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *CoRR*, abs/1607.06450.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In *ACL*, pages 1657–1668.

Shaowei Chen, Jie Liu, Yu Wang, Wenzheng Zhang, and Ziming Chi. 2020a. Synchronous double-channel recurrent network for aspect-opinion pair extraction. In *ACL*, pages 6515–6524.

Xiao Chen, Changlong Sun, Jingjing Wang, Shoushan Li, Luo Si, Min Zhang, and Guodong Zhou. 2020b. Aspect sentiment classification with document-level sentiment preference modeling. In *ACL*, pages 3667–3677.

Zhuang Chen and Tieyun Qian. 2020a. Relation-aware collaborative learning for unified aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 3685–3694.

Zhuang Chen and Tieyun Qian. 2020b. Relation-aware collaborative learning for unified aspect-based sentiment analysis. In *ACL*.

Yang Deng, Wenxuan Zhang, and Wai Lam. 2020. Opinion-aware answer generation for review-driven question answering in e-commerce. In *CIKM '20*, pages 255–264.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2017. An unsupervised neural attention model for aspect extraction. In *ACL*, pages 388–397.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2019. An interactive multi-task learning network for end-to-end aspect-based sentiment analysis. In *ACL19*, pages 504–515.

Guangyi Hu, Chongyang Shi, Shufeng Hao, and Yu Bai. 2020. Residual-duet network with tree dependency representation for chinese question-answering sentiment analysis. In *SIGIR*, page 1725–1728.

Minghao Hu, Yuxing Peng, Zhen Huang, Dongsheng Li, and Yiwei Lv. 2019. Open-domain targeted sentiment analysis via span-based extraction and classification. In *ACL*, pages 537–546.

Kun Li, Chengbo Chen, Xiaojun Quan, Qing Ling, and Yan Song. 2020. Conditional augmentation for aspect term extraction via masked sequence-to-sequence generation. In *ACL*, pages 7056–7066.

Xin Li, Lidong Bing, Piji Li, and Wai Lam. 2019a. A unified model for opinion target extraction and target sentiment prediction. In *AAAI*, pages 6714–6721.

Xin Li, Lidong Bing, Wenxuan Zhang, and Wai Lam. 2019b. Exploiting BERT for end-to-end aspect-based sentiment analysis. In *W-NUT@EMNLP*, pages 34–41.

Yunlong Liang, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2020. An iterative knowledge transfer network with routing for aspect-based sentiment analysis. *CoRR*, abs/2004.01935.

Bing Liu. 2012. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies.

Huaishao Luo, Tianrui Li, Bing Liu, and Junbo Zhang. 2019. DOER: dual cross-shared RNN for aspect term-polarity co-extraction. In *ACL*, pages 591–601.

Yue Mao, Yi Shen, Chao Yu, and Longjun Cai. 2021. A joint training dual-mrc framework for aspect based sentiment analysis. *CoRR*, abs/2101.00816.

Margaret Mitchell, Jacqui Aguilar, Theresa Wilson, and Benjamin Van Durme. 2013. Open domain targeted sentiment. In *EMNLP*, pages 1643–1654.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *SemEval@NAACL-HLT*, pages 486–495.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *SemEval@COLING 2014*, pages 27–35.

Chenlin Shen, Changlong Sun, Jingjing Wang, Yangyang Kang, Shoushan Li, Xiaozhong Liu, Luo Si, Min Zhang, and Guodong Zhou. 2018. Sentiment classification towards question-answering with hierarchical matching network. In *EMNLP*, pages 3654–3663.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. In *NAACL-HLT*, pages 380–385.

Hao Tang, Donghong Ji, Chenliang Li, and Qiji Zhou. 2020. Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. In *ACL*, pages 6578–6588.

Stéphan Tulkens and Andreas van Cranenburgh. 2020. Embarrassingly simple unsupervised aspect extraction. In *ACL*, pages 3182–3187.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.

Jingjing Wang, Changlong Sun, Shoushan Li, Xiaozhong Liu, Luo Si, Min Zhang, and Guodong Zhou. 2019. Aspect sentiment classification towards question-answering with reinforced bidirectional attention network. In *ACL*, pages 3548–3557.

Zhenkai Wei, Yu Hong, Bowei Zou, Meng Cheng, and Jianmin Yao. 2020. Don't eclipse your arts due to small discrepancies: Boundary repositioning with a pointer network for aspect extraction. In *ACL*, pages 3678–3684.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. In *NAACL-HLT*, pages 2324–2335.

Runqi Yang, Jianhai Zhang, Xing Gao, Feng Ji, and Haiqing Chen. 2019. Simple and effective text matching with richer alignment features. In *ACL*, pages 4699–4709.

Lei Zhang, Shuai Wang, and Bing Liu. 2018. Deep learning for sentiment analysis: A survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, 8(4).

Wenxuan Zhang, Yang Deng, and Wai Lam. 2020a. Answer ranking for product-related questions via multiple semantic relations modeling. In *ACM SIGIR*, pages 569–578.

Wenxuan Zhang, Wai Lam, Yang Deng, and Jing Ma. 2020b. Review-guided helpful answer identification in e-commerce. In *WWW '20*, pages 2620–2626.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2021. Towards generative aspect-based sentiment analysis. In *ACL/IJCNLP 2021*, pages 504–510.

He Zhao, Longtao Huang, Rong Zhang, Quan Lu, and Hui Xue. 2020. Spanmlt: A span-based multi-task learning framework for pair-wise aspect and opinion terms extraction. In *ACL*, pages 3239–3248.

Yaowei Zheng, Richong Zhang, Samuel Mensah, and Yongyi Mao. 2020. Replicate, walk, and stop on syntax: An effective neural network model for aspect-level sentiment classification. In *AAAI*, pages 9685–9692.