Fine-Grained Question-Answer Matching via Sentence-Aware Contrastive Self-supervised Transfer

Jingjing Wang, Jiamin Luo, and Guodong Zhou(✉)

School of Computer Science and Technology, Soochow University, Suzhou, China
{djingwang,gdzhou}@suda.edu.cn, 20204027003@stu.suda.edu.cn

Abstract. Previous studies always consider the question-answer (QA) matching task as a one-to-one text matching problem. This study builds upon existing research by expanding the scope to a many-to-many mapping scenario and proposes a new fine-grained QA matching (FQAM) task, which aims to accurately predict the many-to-many matching relationship between all sub-questions and sub-answers within each QA pair. Particularly, a meticulously annotated corpus of high quality is constructed specifically for FQAM to facilitate this research. On this basis, owing to the challenge of expensive data annotation associated with FQAM, we propose a sentence-aware contrastive self-supervised transfer (SCST) approach to transfer the sentence alignment information pre-trained with massive unannotated QA pairs to assist in FQAM. Experimental evaluations conducted on our annotated corpus demonstrate the importance of utilizing sentence alignment information from unannotated QA pairs in FQAM and justify the effectiveness of our approach in capturing such information.

Keywords: Fine-grained Question-Answer Matching · Sentence Alignment Information · Contrastive Self-supervised Learning

1 Introduction

Question-answer (QA) matching plays a crucial role in the fields of NLP and has garnered significant attention due to its diverse applications spanning from fundamental technology services to business intelligence, such as reading comprehension [1,2] and intelligent agent [3,4], which focuses on predicting the one-to-one relationship (*matching* or *non-matching*) between a given QA text pair.

In real-life communications, it is worth noting that individuals often pose multiple (not just one) questions simultaneously, while answer providers are accustomed to answering each of these questions individually. Especially, such phenomenon (e.g., the QA pair with the many-to-many style shown in Fig. 1) is rather pervasive in e-commerce platforms, e.g., *Amazon*, since potential customers tend to ask multiple questions w.r.t. the different aspects of products

F. Liu et al. (Eds.): NLPCC 2023, LNAI 14302, pp. 616–628, 2023.
https://doi.org/10.1007/978-3-031-44693-1_48

A QA Pair with the Many-to-many Style

Q: Is this phone running quickly enough? [Q1] How about the battery capacity? [Q2]

A: The battery capacity has grown by 33%. [A1] But, the speed is disappointing. [A2] I still enjoy the appearance anyway. [A3]

Fine-grained QA Matching

- **Input:** A QA Text Pair
- **Output:** Q1 $\xrightarrow{\times}$ A1; Q1 $\xrightarrow{\checkmark}$ A2; Q1 $\xrightarrow{\times}$ A3; Q2 $\xrightarrow{\checkmark}$ A1; Q2 $\xrightarrow{\times}$ A2; Q2 $\xrightarrow{\times}$ A3.

Fig. 1. An example for illustrating the proposed fined-grained QA matching (FQAM) task, where ✓ and ✗ denote *matching* and *non-matching* respectively.

before making the purchase decisions. Unfortunately, all sub-questions and sub-answers of a single-turn QA are usually integrated into a single question text and a single answer text respectively, thereby making the traditional one-to-one QA matching inapplicable. To this end, we propose a new fine-grained QA matching (FQAM) task to recognize the many-to-many mapping relationship between all sub-questions and sub-answers within each QA text pair.

Unlike the traditional one-to-one QA matching virtually not requiring manual annotation, the data annotation for FQAM is rather expensive since the many-to-many relationship in each QA pair needs to be manually annotated sentence by sentence. Thus, the data scale of FQAM is limited, inevitably restricting the performance. Inspired by recent pre-trained language models (e.g., BERT [5]) having achieved SOTA performances in many text matching tasks, a potential solution to remedy the above issue is to further train the pre-trained models (e.g., BERT) with the large-scale and easy-obtained unannotated QA pairs, and then fine-tune[1] BERT to help FQAM. Despite this, we believe this solution still fails to control two kinds of key information flows for FQAM, i.e., *sentence alignment* and *contrastive noise* information.

For one thing, prior pre-trained models (e.g., BERT) consistently neglect to explicitly capture the *sentence alignment* information between QA, while this information is rather crucial for FQAM. Take Fig. 1 as an example, sub-question **Q1** is exactly located in the first sentence of the question and the aligned sub-answer **A2** is exactly located in the second sentence of answer. Inspired by this, self-supervisedly learning *sentence alignment* information with massive unannotated QA pairs may powerfully contribute to aligning sub-questions with sub-answers. Thus, a better-behaved pre-trained model for FQAM should incorporate the sentence alignment information between QA during pre-training with unannotated QA pairs.

For another, to learn the sentence alignment information between QA, a feasible way is to mask the sentence and then recover it with some pretext tasks

[1] Except for further pre-training BERT, we can also use these unannotated QA pairs to perform a simple one-to-one QA matching task for fine-tuning BERT, i.e., the baselines with "+ QA Pair" shown in Table 2.

like word and span masking [5,6]. However, when processing the text matching tasks, BERT and its variants always leverage a mark "[SEP]" to concatenate the QA pair into a sequence for implicitly learning the QA matching information. This is suboptimal for FQAM since the sequence auto-regressive property may bring much noise from the question when we recover the masked sentence in the answer, and vice versa. Take Fig. 1 as an example, when we mask sub-answer **A2**, only sub-question **Q1** contributes to recovering **A2**. In contrast, the nearer sub-question **Q2** is noisy and should be filtered as much as possible. In this study, this noisy sub-question **Q2** is defined as the *contrastive noise* for sub-answer **A2**. Obviously, treating QA as a sequence cannot filter this noise well. We believe a better-behaved pre-trained model for FQAM should treat the QA pair as two parallel units and explicitly filter this noise.

In this paper, we propose a new two-step self-supervised learning framework, namely Sentence-aware Contrastive Self-supervised Transfer (SCST) approach, to tackle the above two challenges. In the first step, we propose a sentence-aware contrastive self-supervised learning model for pre-training the massive unannotated QA pairs with two pretext tasks (i.e., sentence retrieval and generation). Wherein, a novel contrastive bidirectional attention encoder is designed to capture the *sentence alignment* information between QA and meantime filter the aforementioned *contrastive noise*. In the second step, we propose a sequence decoding model to perform FQAM, and transfer the fine-tuned parameters of the above pre-trained contrastive bidirectional attention encoder to initialize the QA pair encoder inside this sequence decoding model for performing FQAM. Experimental results demonstrate the impressive effectiveness of the SCST approach to FQAM over the SOTA baselines.

2 Approach

In this section, we introduce our SCST approach, the framework of which is shown in Fig. 2. Similar to the prior self-supervised frameworks [14], SCST also consists of two steps. The first step is a sentence-aware contrastive self-supervised learning model for pre-training the massive unannotated QA pairs. The second step is a sequence decoding model for performing FQAM, aiming to transfer the pre-trained knowledge of the first step to boost the FQAM performance.

2.1 Step1: Sentence-Aware Contrastive Self-supervised Learning

Given an unannotated QA text pair $[\mathbf{Q}, \mathbf{A}]$, we first adopt sentence segmentation tool[2] to segment question and answer texts into sentence sequences $\{\mathbf{q}_1, ..., \mathbf{q}_m\}$ and $\{\mathbf{a}_1, ..., \mathbf{a}_n\}$ respectively. Then, we leverage a mark "[SEN]" to distinguish and concatenate the sentences of each question or answer in the following way:

$$\begin{aligned} \mathbf{Q}&: \ \mathbf{q}_1 \ [\texttt{SEN}] \ \mathbf{q}_2 \ [\texttt{SEN}] \ldots \ \mathbf{q}_m \ [\texttt{SEN}] \\ \mathbf{A}&: \ \mathbf{a}_1 \ [\texttt{SEN}] \ \mathbf{a}_2 \ [\texttt{SEN}] \ldots \ \mathbf{a}_n \ [\texttt{SEN}] \end{aligned} \tag{1}$$

[2] http://stanfordnlp.github.io/CoreNLP.

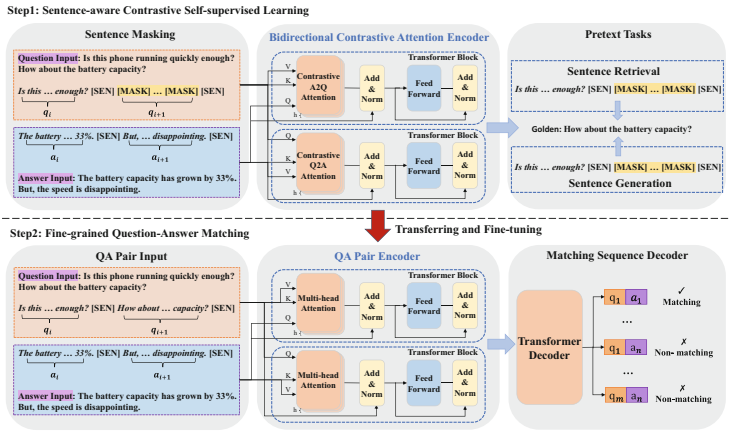

Fig. 2. The framework of our Sentence-aware Contrastive Self-supervised Transfer (SCST) approach to FQAM.

Contrastive Bidirectional Attention Encoder is proposed to unsupervisedly learn the sentence alignment information between QA, which is then transferred to assist the downstream FQAM. This encoder is delicately designed by modifying the basic transformer which is reviewed as follows.

• **Basic Transformer Block** proposed by Vaswani et al. [16] aims to leverage an h-head self-attention to transform each position in the input sequence into a weighted sum of the input sequence itself. Specifically, for each head attention, given an input sequence $\mathbf{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n)$, this sequence $\mathbf{X}$ is first transformed according to the self-attention as:

$$\text{selfAtt}(\mathbf{X}, \mathbf{X}) = \text{softmax}(\frac{\mathbb{Q}_X^\top \mathbb{K}_X}{\sqrt{d_k}})\mathbb{V}_X \tag{2}$$

where $\mathbb{Q}_X = \mathbf{W}_Q\mathbf{X}$, $\mathbb{K}_X = \mathbf{W}_K\mathbf{X}$ and $\mathbb{V}_X = \mathbf{W}_V\mathbf{X}$ corresponds to queries, keys and values respectively. $\mathbf{W}_Q$, $\mathbf{W}_K$ and $\mathbf{W}_V$ are trainable parameters. $\sqrt{d_k}$ is the scaling factor. Second, all outputs of the h head attentions are concatenated as the following formula: $\text{multiAtt}(\mathbf{X}, \mathbf{X}) = \mathbf{W}_h[\text{selfAtt}_1(\cdot), ..., \text{selfAtt}_h(\cdot)]^\top$, where $\mathbf{W}_h$ is a trainable parameter. Then a residual connection followed by a normalization operation is used to obtain the further representation as: $\mathbf{H} = \mathbf{LN}(\mathbf{X} + \text{multiAtt}(\mathbf{X}))$. Finally, the output of transformer block is computed as follows:

$$\mathbf{R} = \mathbf{LN}\big(\mathbf{X} + \mathbf{FFN}(\mathbf{H})\big) \tag{3}$$

where $\mathbf{LN}(\cdot)$, $\mathbf{FFN}(\cdot)$ represent layer normalization and feed forward network.

• **Contrastive Bidirectional Attention** is designed to compute contrastive answer-to-question (A2Q) and question-to-answer (Q2A) attentions.

(1) Contrastive A2Q Attention. This attention aims to highlight the words in question aligned with masked sentence $\mathbf{a}_i$ in answer, while filter the con-

trastive noise in question. Given a QA pair $[\mathbf{Q}, \mathbf{A}]$ which has been processed by Eq. (1), and suppose that $\mathbf{A}$ has been performed sentence masking and masked into the new answer $\hat{\mathbf{A}}$, the contrastive A2Q attention $\text{coAtt}(\hat{\mathbf{A}}, \mathbf{Q})$ is then computed by modifying Eq. (2) in a transformer block as follows:

$$\text{coAtt}(\hat{\mathbf{A}}, \mathbf{Q}) = \big(\mathbb{1} - \text{softmax}(\frac{\mathbb{Q}_{\hat{A}}^{\top}\mathbb{K}_Q}{\sqrt{d_k}})\big)\mathbb{V}_Q \tag{4}$$

where $\mathbb{Q}_{\hat{A}} = \mathbf{W}_{Q}\hat{\mathbf{A}}$, $\mathbb{K}_Q = \mathbf{W}_{K}\mathbf{Q}$ and $\mathbb{V}_Q = \mathbf{W}_{V}\mathbf{Q}$. $\mathbb{1}$ is a unit matrix whose values are all 1. $\mathbb{1} - \text{softmax}(\cdot)$ denotes an operation of calculating opponent attention weights for all words in question. With this operation, we can highlight the part of question dissimilar to unmasked sentences in $\hat{\mathbf{A}}$, since we assume that this part is aligned with the masked sentence $\mathbf{a}_i$. Then, a purified question matrix $\mathbf{R}^{(q)}$ is computed by Eq. (3) for recovering masked answer sentence $\mathbf{a}_i$.

(2) Contrastive Q2A Attention. This attention aims to highlight the words in answer aligned with masked sentence $\mathbf{q}_i$ in question, while filter the corresponding contrastive noise. Given a QA pair $[\mathbf{Q}, \mathbf{A}]$ which has been processed by Eq. (1), and suppose that $\mathbf{Q}$ has been performed sentence masking and masked into the new answer $\hat{\mathbf{Q}}$, contrastive Q2A attention $\text{coAtt}(\hat{\mathbf{Q}}, \mathbf{A})$ is computed by modifying Eq. (2) in another transformer block as:

$$\text{coAtt}(\hat{\mathbf{Q}}, \mathbf{A}) = \big(\mathbb{1} - \text{softmax}(\frac{\mathbb{Q}_{\hat{Q}}^{\top}\mathbb{K}_A}{\sqrt{d_k}})\big)\mathbb{V}_A \tag{5}$$

where $\mathbb{Q}_{\hat{Q}} = \mathbf{W}_{Q}\hat{\mathbf{Q}}$, $\mathbb{K}_A = \mathbf{W}_{K}\mathbf{A}$ and $\mathbb{V}_A = \mathbf{W}_{V}\mathbf{A}$.

Then, a purified answer matrix $\mathbf{R}^{(a)}$ is computed by Eq. (3) for recovering masked question sentence $\mathbf{q}_i$. Note that, since every time only one side in a QA pair is masked, operation $\mathbb{1} - \text{softmax}(\cdot)$ will not be performed in both A2Q and Q2A attention simultaneously.

Pretext Tasks. In our SCST approach, we design sentence retrieval and sentence generation two pretext tasks for pre-training.

• **Sentence Retrieval** aims to select the correct masked sentence from a set of k candidate sentences. Specifically, we first run each sentence independently through a transformer block and obtain the matrix $\mathbf{R}$ of a sentence by Eq. (3). Then, we use a max-pooling to compute the sentence vector as $\boldsymbol{r}_i = \text{pooling}(\mathbf{R})$. Given the sentence set $\mathbf{r} = \{\boldsymbol{r}, ..., \boldsymbol{r}_l\}$ where l is the number of sentences in all answer texts, this pretext task is to select the masked sentence $\boldsymbol{r}_i$ from this set $\mathbf{r}$. Note that $\mathbf{r}$ is usually very large and a more computationally feasible way is to sample a subset of $\mathbf{r}$ and thus we retrieve negative samples for each masked sentence according to the uniform distribution [12]. Subsequently, we concatenate the question matrix $\mathbf{R}^{(q)}$ and the answer matrix $\mathbf{R}^{(a)}$ to compute the final vector of the masked QA pair as $\boldsymbol{s}_i = \mathbf{W}_s[\text{pooling}(\mathbf{R}^{(q)}); \text{pooling}(\mathbf{R}^{(a)})]$. Here, $\mathbf{W}_s$ is the trainable parameter and $[;]$ denotes the vector concatenation. Finally, the cross-entropy loss of retrieving the masked sentence is given by:

$$\mathcal{L}^{(r)} = -\log p(\boldsymbol{r}_i|\boldsymbol{r}_{1:l}) = -\log(\frac{\exp(\boldsymbol{s}_i{}^\top \boldsymbol{r}_i)}{\sum_{j=1}^{k} \exp(\boldsymbol{s}_j{}^\top \boldsymbol{r}_j)}) \tag{6}$$

• **Sentence Generation** aims to generate the masked sentence token by token. For clarity, we take the generation of masked sentence $\boldsymbol{r}_i$ in answer side as an example. Specifically, we adopt the text generation approach proposed by Mehri et al. [12] to generate the masked sentence $\boldsymbol{r}_i$. Then, let the tokens in $\boldsymbol{r}_i$ be $[w_1, ..., w_N]$, the related likelihood loss of generating $\boldsymbol{r}_i$ is defined as:

$$\mathcal{L}^{(g)} = -\log p(\boldsymbol{r}_i|\boldsymbol{r}_{1:l}) = -\sum_{j}^{N} \log p(w_j|w_{<j}, \boldsymbol{s}_i) \tag{7}$$

where $\boldsymbol{s}_i$ is the final vector of the masked QA pair.

Table 1. Statistics of our constructed datasets. #s/Q (#s/A) and #ch/Q (#ch/A) denote the average number of sentences and Chinese characters in each question (answer) respectively. #m and #n denote the number of *matching* and *non-matching* sentence pairs.

Datasets	#QA	#s/Q	#s/A	#ch/Q	#ch/A	#m	#n
Annotated	32k	3.1	3.9	20.2	28.7	115k	272k
Unannotated	500k	3.4	4.3	21.2	31.2	–	–

2.2 Step2: Fine-Grained QA Matching

In the second step, we formulate FQAM as follows. Given a QA text pair $[\mathbf{Q}, \mathbf{A}]$ where $\mathbf{Q}$ is a sub-question sequence $\{\mathbf{q}_1, ..., \mathbf{q}_m\}$ and $\mathbf{A}$ is a sub-answer sequence $\{\mathbf{a}_1, ..., \mathbf{a}_n\}$, we construct a matching sequence of sub-question and sub-answer pair $[\mathbf{q}_i, \mathbf{a}_j]_{i=1,j=1}^{m,n}$ and obtain mn pairs in total. The goal of FQAM is to predict each pair $\mathcal{P}_t = [\mathbf{q}_i, \mathbf{a}_j]$, $t \in [1, mn]$ inside this matching sequence is *matching* or *non-matching*, which could be seen as a sequence labeling problem. In this way, we design a sequence decoding model to perform FQAM, consisting of a QA pair encoder and a matching sequence decoder as illustrated in Fig. 2.

QA Pair Encoder. To achieve the goal of using the unannotated QA pairs to help FQAM, we transfer and fine-tune the parameters of the contrastive bidirectional encoder pre-trained in the first step to initialize the QA pair encoder in the second step. Suppose that a QA pair $[\mathbf{Q}, \mathbf{A}]$ has been processed by Eq. (1), we first feed this pair to the QA pair encoder. Then, we directly treat the output vector of the closely followed mark "[SEN]" as the vector $\boldsymbol{c}_i^{(q)}$ of sub-question $\mathbf{q}_i$, while the vector $\boldsymbol{c}_j^{(a)}$ is for sub-answer $\mathbf{a}_j$, since transformer possesses the auto-regressive property. Further, we construct the vector-pair

sequence $[\boldsymbol{c}_i^{(q)}, \boldsymbol{c}_j^{(a)}]_{i=1,j=1}^{m,n}$ for all sub-question and sub-answer pairs and feed it to the next matching sequence decoder.

Matching Sequence Decoder. Once obtained the vector-pair sequence $[\boldsymbol{c}_i^{(q)}, \boldsymbol{c}_j^{(a)}]_{i=1,j=1}^{m,n}$, the two vectors inside each vector-pair are concatenated before feeding this sequence into a transformer decoder. Then, a max-pooling is used to transform the transformer output into the final vector $\boldsymbol{v}_t$ for each sub-question and sub-answer pair $\mathcal{P}_t$. Finally, we feed the vector $\boldsymbol{v}_t$ of pair $\mathcal{P}_t$ to a softmax layer for computing the final probabilities of different labels as $p(y|\mathcal{P}_t)$, $y \in \{matching, non\text{-}matching\}$.

2.3 Model Training

In the literature, existing self-supervised learning models [14,17] often adopt the multi-task learning pre-training procedure. In the first step, our contrastive self-supervised model also uses the multi-task learning framework to jointly learn the sentence retrieval and generation tasks.

Then, the joint loss function for our self-supervised model in the first step is defined as: $\mathcal{L} = \mathcal{L}^{(r)} + \mathcal{L}^{(g)}$, where $\mathcal{L}^{(r)}$ and $\mathcal{L}^{(g)}$ are the losses for sentence retrieval and sentence generation respectively. In the second step, with the label probabilities of each sub-question and sub-answer pair $\mathcal{P}_t$, we minimize the negative log-likelihood loss of a QA text pair for the FQAM task as: $\mathcal{L}^{(f)} = -\sum_{t=1}^{T} \log p(y_t|\mathcal{P}_t)$, where $T = mn$ is the number of pair $\mathcal{P}_t$ in a QA text pair. y_t is the ground-truth label for $\mathcal{P}_t$.

3 Experimental Settings

Data Settings. In this study, we collect 532k Chinese QA text pairs from the e-commerce platform *Taobao*[3] on which the many-to-many style QA pairs are rather pervasive. Then, we randomly select 32k QA text pairs to perform manual annotation for FQAM, and treat the rest 500k QA pairs as the large-scale unannotated dataset for performing pre-training. Note that, we adopt the well-studied sentence segmentation tool[2] to segment each question and answer text into sentences and then directly treat each sentence as the sub-question or sub-answer to perform annotation for FQAM. We believe the above process can make FQAM more practical to large-scale application and can also make our SCST approach more flexible and scalable, since most of sub-questions and sub-answers are located in one sentence and more importantly deciding the boundary of each sub-question and sub-answer is rather difficult and not practical for real-world applications. The statistics of our constructed datasets are shown in Table 1. Then, we randomly split the ***Annotated*** dataset into train, dev, and test sets with the ratio of 7:1:2 for FQAM.

[3] http://www.taobao.com.

Table 2. Performance comparison of various approaches to FQAM, where P, R, F1 and Acc. represent Precision, Recall, Macro-F1 and Accuracy, respectively. Besides, "+ QA Pairs" denotes BERT and RoBERTa are further trained with our 500k unannotated QA pairs, and then fine-tuned by casting these unannotated QA pairs as a one-to-one QA matching task.

	Approaches	P	R	F1	Acc.
Single	BiMPM [7]	68.3	69.2	68.7	69.4
	ESIM [18]	68.7	68.9	68.8	71.2
	RE2 [19]	68.9	72.3	70.6	71.7
	BERT [5]	73.1	74.7	73.9	75.1
	BERT + QA Pairs	75.2	76.8	76.0	77.5
	RoBERTa [15]	73.9	75.9	74.9	76.3
	RoBERTa + QA Pairs	76.3	77.5	76.9	77.1
Sequence	BiMPM [7]	69.7	70.2	69.9	71.1
	ESIM [18]	71.1	71.7	71.4	71.5
	RE2 [19]	71.7	73.7	72.7	73.8
	BERT [5]	74.5	76.2	75.3	76.5
	BERT + QA Pairs	75.3	77.8	76.5	77.2
	RoBERTa [15]	74.8	77.4	76.1	78.1
	RoBERTa + QA Pairs	75.2	**79.5**	77.3	78.3
Ours	SCST (only using Step2)	72.3	74.2	73.2	74.1
	SCST	**78.3**	79.2	**78.7**	**80.5**

Baselines. For FQAM, we implement some approaches [5,7,15,18,19] as baselines. Concretely, the baselines contain **Single** and **Sequence** two groups. **Single** group is to treat the matched sub-question and sub-answer pair as positive samples while the mismatched pairs as negative samples and then directly use the baseline approach to perform binary classification for each sample. **Sequence** group is to treat FQAM as a sequence labeling problem. Specifically, we first use a shared encoder to encode each sub-question and sub-answer pair, and then use an extra transformer decoder to decode matching sequence like our SCST approach. Here, different approaches shown in Table 2 are adopted as the shared encoder for comparison.

Implementation Details. All experiments adopt BERT-Base (Chinese) and RoBERTa-Base (Chinese). In the first step of SCST, contrastive bidirectional attention encoder is first initialized with the last layer of the released RoBERTa, and then pre-trained with 500k unannotated pairs. Besides, in each experiment, we measure the runtime 10 times and average the results.

Evaluation Metrics. The performance is evaluated with *Precision* (P), *Recall* (R), *Macro-F1* (F1) and *Accuracy* (Acc.). Besides, the paired t-test[4] is used to evaluate the significance of the performance difference of two approaches.

4 Results and Discussion

Experimental Results. Table 2 shows the performances of different approaches to FQAM. From this table, we can see that, two large-scale pre-trained self-supervised models, i.e., **BERT** and **RoBERTa**, significantly outperform (p-value < 0.05) the traditional attention based text matching approaches, i.e., **BiMPM**, **ESIM** and **RE2**. This confirms the powerful transfer ability of pre-trained models for downstream tasks including FQAM. When further trained and fine-tuned with unannotated QA pairs, **BERT** and **RoBERTa** can consistently achieve better performance. This encourages us to leverage the large-scale unannotated QA pairs to boost the performance of FQAM whose annotation is rather time-consuming and labor-intensive.

Furthermore, when treating FQAM as a sequence labeling problem, all approaches in the second group **Sequence** perform better than their corresponding approaches in the first group **Single**. This is reasonable. Take Fig. 1 as an example, if Q2 has been predicted to be matched with A1, the pair Q2 and A3 is more possibly to be predicted as *non-matching*. This dependency information could be well captured by the transformer decoder. These results encourage us to treat FQAM as a sequence labeling problem instead of binary classification.

In contrast, our **SCST** approach outperforms all above baselines and even significantly outperforms (p-value < 0.05) the strong pre-trained model **RoBERTa + QA Pairs**. This justifies the effectiveness of SCST in unsupervisedly capturing the sentence alignment information between QA. Impressively, compared with **SCST** which removes Step1, **SCST** achieves the improvement of 5.5% in terms of *Macro-F1* and 6.4% in terms of *Accuracy*. Significance test shows that these improvements are significant (p-value < 0.05). This again justifies the importance of leveraging the unannotated QA pairs to help FQAM and the effectiveness of SCST in performing pre-training with these QA pairs.

Ablation Study. Table 3 shows the ablation results to evaluate the contribution of each key component for SCST. From Table 3, we can see that: **1)** Fine-tuning bidirectional contrastive attention encoder with the annotated FQAM dataset can improve the Acc. by 1.4%. **2)** Incorporating contrastive Q2A and A2Q attention into SCST can improve the Acc. by 2.3% and 1.2%. This indicates that treating QA text pairs as parallel units and considering the bidirectional matching information is helpful. **3)** Using the pretext task sentence retrieval and sentence generation can improve the Acc. by 1.8% and 1.2%. **4)** Using RoBERTa to initialize the transformer blocks inside contrastive bidirectional encoder can improve the Acc. by 3.3%. This is reasonable since the data for pre-training RoBERTa from scratch is much larger than our 500k QA pairs.

[4] https://www.scipy.org/.

Table 3. Ablation study for our SCST approach.

Approaches	P	R	F1	Acc.
SCST	**78.3**	**79.2**	**78.7**	**80.5**
– Fine-tuning with FQAM	77.5	78.2	77.8	79.1
– Contrastive Q2A Attention	75.3	76.9	76.1	78.2
– Contrastive A2Q Attention	76.5	78.8	77.6	79.3
– Sentence Retrieval	76.9	78.1	77.5	78.7
– Sentence Generation	77.2	78.7	77.9	79.3
– Initialized with RoBERTa	74.3	76.9	75.6	77.2

Table 4. Performance comparison of SCST and four strong baselines on the purified test set where all the QA pairs have the reversed-order phenomenon.

	Approaches	P	R	F1	Acc.
Single	BERT + QA Pairs	63.4	66.2	64.8	65.7
	RoBERTa + QA Pairs	64.3	66.3	65.3	66.8
Sequence	BERT + QA Pairs	67.5	69.1	68.3	70.2
	RoBERTa + QA Pairs	68.2	70.3	69.2	70.6
Ours	**SCST**	**71.9**	**74.3**	**73.1**	**74.2**

Robustness Analysis of Transfer. To investigate the transfer ability of our SCST approach, we evaluate it and three strong baselines on different numbers of training examples for FQAM. As shown in Fig. 3, **1)** SCST can significantly improve the performance from 34.7% to 46.2% on small size of training data (only 10% training data) compared to RE2 without pre-training. **2)** SCST consistently performs better than all pre-training based approaches, i.e., BERT and RoBERTa, on various sizes of training data. These justify the robustness of SCST in terms of transfer to FQAM.

Effectiveness Analysis of Sequence Decoding with Reversed-order Samples. To evaluate the effectiveness of our proposed matching sequence decoder for FQAM and enhance the task difficulty, we construct a cleaned and more difficult test set, of which all the QA pairs have the reversed-order phenomenon (e.g. Q1 $\xrightarrow{\checkmark}$ A2 while Q2 $\xrightarrow{\checkmark}$ A1 in Fig. 1). Table 4 shows the results on the cleaned test set. From which, we can see that the approaches in the **Sequence** group perform much better than **Single**. This justifies the effectiveness of our proposed sequence decoder and encourages us to consider FQAM as a sequence labeling problem. Moreover, our SCST approach still outperforms (p-value < 0.05) all the four strong baselines, again justifying the robustness of SCST.

Error Analysis of SCST. We randomly analyze 200 error cases of SCST and categorize them into 4 types. **1)** 48% of errors are due to the ambiguous

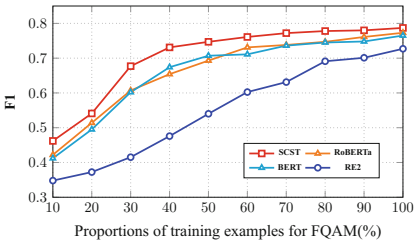

Fig. 3. Comparison of SCST and three baselines from the **sequence** group, trained with different proportions of training examples for FQAM, where BERT and RoBERTa have been further trained and fine-tuned with 500k QA pairs.

reference and complicated semantics. An example is "Q: *How is screen? How about battery? Is the speed fast?* A: *Not good. Anyway, screen is nice. The speed is impressive.*". SCST fails to capture the related sub-question of the sub-answer "*not good*" with the ellipsis of reference; **2)** 26% of errors are due to the lack of external knowledge. An example is "Q: *Can it run Tensorflow?* A: *Don't worry, it has perfect hardware.*". SCST incorrectly predicts *non-matching*. **3)** 14% are due to the long length (e.g., more than 10 sentences) of questions or answers. **4)** 12% are due to fuzzy boundaries and incorrect segmentation of sub-questions or sub-answers. An example is "Q: *How about price, quality and service?* A: *Not expensive. Good quality. But the after-sale service is terrible.*". SCST incorrectly predicts *non-matching* for sub-answer "*not expensive*", inspiring us to perform question/answer decomposition [20] for FQAM.

5 Related Work

Text Matching aims to predict whether a given text pair has similar semantics. Dominant paradigms for text matching focus on leveraging the multi-perspective matching model [7] and attention based neural networks [8] to capture the matching relationship. QA matching is a sub-task of text matching, which aims to predict the matching relationship between a QA pair. Recently, Shen et al. [9] propose to compute the co-occurrence probability in QA pairs. Wang et al. [10] also propose a fine-grained QA matching task, but their task is limited to the one-to-many scenario. Unlikely, we extend text matching to a more general many-to-many matching paradigm and propose a new FQAM task. To our best knowledge, this is the first attempt to address this task.

Self-supervised Learning uses pretext tasks to replace the manually annotated labels with "pseudo-label" obtained from the raw data. Actually, BERT [5] is a typical self-supervised model, adopting the word masking to perform pre-training. Besides, Wu et al. [11] and Mehri et al. [12] leverage utterance masking to detect the utterance order inside dialogue for pre-training. Recently, contrastive learning [13] has attracted much attention, which incorporates negative pairs selection and contrastive losses to perform self-supervised learning.

Unlike all the above studies, we propose a new contrastive self-supervised learning framework, not relying on either negative pairs or contrastive losses, to unsupervisedly capture the sentence alignment information between QA.

6 Conclusion

In this paper, we propose a new FQAM task and build a high-quality annotated corpus for this task. On this basis, we propose a sentence-aware contrastive self-supervised transfer (SCST) approach. The basic idea of SCST is to leverage large-scale unannotated QA pairs to help FQAM with limited labeled data. Empirical studies show that SCST significantly outperforms two SOTA pre-trained baselines in FQAM. In our future work, we would like to solve other challenges in FQAM, such as the ellipsis of reference and the lack of external knowledge. Furthermore, we would like to investigate our SCST approach in other tasks whose inputs are also parallel units, e.g., cross-lingual analysis.

Acknowledgements. This work was supported by three NSFC grants, i.e., No. 62006166, No. 62076176 and No. 61976146. This work was also supported by a Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD). Also, we would like to thank the anonymous reviewers for their helpful comments.

References

1. Trischler, A., Ye, Z., Yuan, X., He, J., Bachman, P.: A parallel-hierarchical model for machine comprehension on sparse data. In: Proceedings of ACL 2016, Berlin, Germany (2016)
2. Yang, Z., et al.: HotpotQA: a dataset for diverse, explainable multi-hop question answering. In: Proceedings of EMNLP 2018, Brussels, Belgium, pp. 2369–2380 (2018)
3. Cui, L., Huang, S., Wei, F., Tan, C., Duan, C., Zhou, M.: SuperAgent: a customer service chatbot for E-commerce websites. In: Proceedings of ACL 2017, Vancouver, Canada, pp. 97–102 (2017)
4. Wang, J.C., et al.: Sentiment classification in customer service dialogue with topic-aware multi-task learning. In: Proceedings of AAAI 2020, New York, USA, pp. 9177–9184 (2020)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL 2019, Minneapolis, pp. 4171–4186 (2019)
6. Joshi, M., Chen, D., Liu, Y., Weld, D.S., Zettlemoyer, L., Levy, O.: SpanBERT: improving pre-training by representing and predicting spans. Trans. Assoc. Comput. Linguist. **8**, 64–77 (2020)
7. Wang, Z., Hamza, W., Florian, R.: Bilateral multi-perspective matching for natural language sentences. In: Proceedings of IJCAI 2017, Melbourne, Australia, pp. 4144–4150 (2017)
8. Rao, J., Liu, L., Tay, Y., Yang, H.W., Shi, P., Lin, J.: Bridging the gap between relevance matching and semantic matching for short text similarity modeling. In: Proceedings of EMNLP 2019, China, pp. 5369–5380 (2019)

9. Shen, Y., Rong, W., Jiang, N., Peng, B., Tang, J., Xiong, Z.: Word embedding based correlation model for question/answer matching. In: Proceedings of AAAI 2017, San Francisco, California, USA, pp. 3511–3517 (2017)
10. Wang, L., et al.: One vs. many QA matching with both word-level and sentence-level attention network. In: Proceedings of COLING 2018, New Mexico, USA, pp. 2540–2550 (2018)
11. Wu, J., Wang, X., Wang, W.Y.: Self-supervised dialogue learning. In: Proceedings of ACL 2019, Italy, pp. 3857–3867 (2019)
12. Mehri, S., Razumovskaia, E., Zhao, T., Eskénazi, M.: Pretraining methods for dialog context representation learning. In: Proceedings of ACL 2019, Florence, Italy, pp. 3836–3845 (2019)
13. Yang, Z., Cheng, Y., Liu, Y., Sun, M.: Reducing word omission errors in neural machine translation: a contrastive learning approach. In: Proceedings of ACL 2019, Florence, Italy, pp. 6191–6196 (2019)
14. Wang, S., Che, W., Liu, Q., Qin, P., Liu, T., Wang, W.Y.: Multi-task self-supervised learning for disfluency detection. In: Proceedings of AAAI 2020, New York, NY, USA, pp. 9193–9200 (2020)
15. Liu, Y., et al.: RoBERTa: a robustly optimized BERT pretraining approach. CoRR (2019)
16. Vaswani, A., et al.: Attention is all you need. In: Proceedings of NeurIPS 2017, Long Beach, CA, pp. 5998–6008 (2017)
17. Doersch, C., Zisserman, A.: Multi-task self-supervised visual learning. In: Proceedings of ICCV 2017, Venice, Italy, pp. 2070–2079 (2017)
18. Chen, Q., Zhu, X., Ling, Z.H., Wei, S., Jiang, H., Inkpen, D.: Enhanced LSTM for natural language inference. In: Proceedings of ACL 2017, Vancouver, Canada, pp. 1657–1668 (2017)
19. Yang, R., Zhang, J., Gao, X., Ji, F., Chen, H.: Simple and effective text matching with richer alignment features. In: Proceedings of ACL 2019, Florence, Italy, pp. 4699–4709 (2019)
20. Perez, E., Lewis, P.S.H., Yih, W., Cho, K., Kiela, D.: Unsupervised question decomposition for question answering. CoRR (2020)