1

Responding E-commerce Product Questions via Exploiting QA Collections and Reviews

Qian Yu, Wai Lam, Zihao Wang
Department of Systems Engineering and Engineering Management
The Chinese University of Hong Kong
{yuqian, wlam, zhwang}@se.cuhk.edu.hk

Abstract

Providing instant responses for product questions in E-commerce sites can significantly improve satisfaction of potential consumers. We propose a new framework for automatically responding product questions newly posed by users via exploiting existing QA collections and review collections in a coordinated manner. Our framework can return a ranked list of snippets serving as the automated response for a given question, where each snippet can be a sentence from reviews or an existing question-answer pair. One major subtask in our framework is question-based response review ranking. Learning for response review ranking is challenging since there is no labeled response review available. The collection of existing QA pairs are exploited as distant supervision for learning to rank responses. With proposed distant supervision paradigm, the learned response ranking model makes use of the knowledge in the QA pairs and the corresponding retrieved review lists. Extensive experiments on datasets collected from a real-world commercial E-commerce site demonstrate the effectiveness of our proposed framework.

1 Introduction

Many E-commerce sites provide product information on product pages. Under a particular product page, users can ask questions about the corresponding product. Other experienced or qualified users can voluntarily provide answers. Thus, for each product, a product-specific question-answer (QA) collection is commonly available in E-commerce sites. However, the amount of QA pairs in the QA collection associated with a product is small since it is product-specific. In addition to the QA collection, users may write reviews about the product and a product-specific review collection is also commonly available. Figure 1 depicts an example, from a real-world E-commerce site, of a product page of a bluetooth car device. In this product page, the middle region contains the question-answer collection. The bottom region contains the review collection. Users may browse those two information sources for obtaining useful or insightful information about the corresponding product.

Both information sources mentioned above would be helpful for providing instant responses to questions newly posed by users about the corresponding product. Consider a new question "Is this device compatible with 2004 VW Jetta SportWagen?", a semantically similar question can be found in the existing QA collection of this product. Specifically the first question is a good match with the new question. Thus, the user posed answer for that question can be extracted as the response. On the other hand, consider another new question "Can I use iPhone 4s with it?". This new question cannot match any existing question well. In spite of that, the review sentence "My iPhone 4S connected with the unit and uploaded my contacts with no issues." found in the review collection of the corresponding product can provide informative content which can be retrieved as a response.

In this paper, we investigate the problem of responding E-commerce product questions newly posed by users. We aim at providing responses before any answer is posed by other users. Such kind of instant responses can improve user satisfaction. As aforementioned, two commonly available information

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: 14203414).

Proceedings of the 27th International Conference on Computational Linguistics, pages 2192–2203
Santa Fe, New Mexico, USA, August 20-26, 2018.

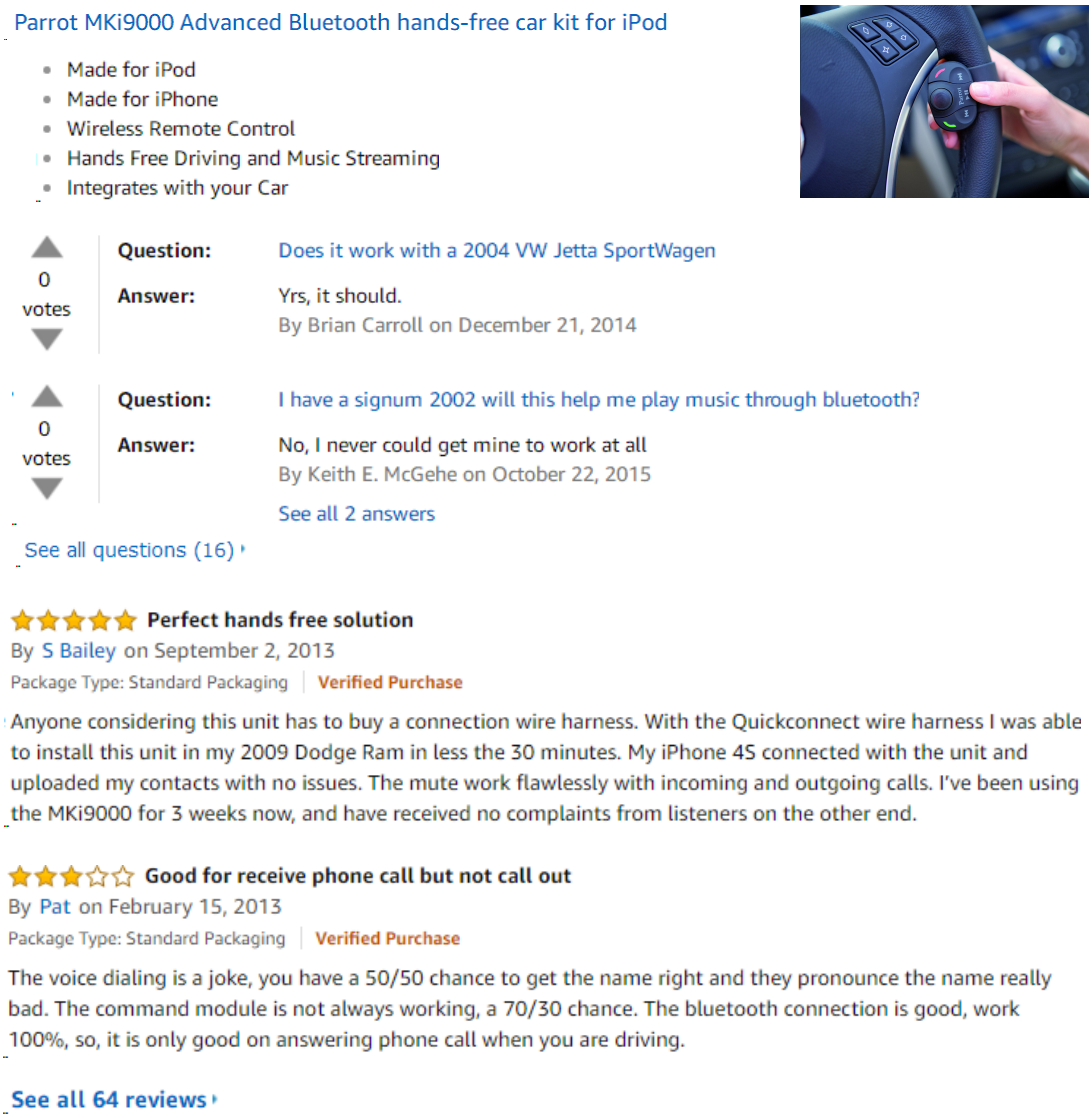

Figure 1: A product page of a bluetooth car device.

sources, namely, question-answer collections and review collections contain valuable information for generating instant responses for product questions. Consequently, given a question, our framework can return a ranked list of snippets serving as the automated response, where each snippet can be a sentence from reviews or an existing question-answer pair.

Community Question Answering (CQA) approaches can be employed, but they can only make use of the QA collection of the corresponding product, which typically contains just a small amount of QA pairs. Furthermore, similar questions associated with other products are not helpful due to different product specifications. Another limitation of CQA methods is that they cannot make use of the review collection which is another important information source for generating responses. Another possible approach is to employ a question answering learning approach such as QA-LSTM (Tan et al., 2016). But these QA models typically cannot effectively exploit reviews due to the heterogeneous nature of answer collections and review collections. Recently a chatbot for E-commerce sites known as SuperAgent has been developed (Cui et al., 2017). SuperAgent considers both QA collections and reviews when answering questions. However, it employs separated modules for each of the information sources without mutual coordination. Moreover, it requires external knowledge and a large volume of annotated data. Some models based on Mixture of Expert (MoE) (McAuley and Yang, 2016; Wan and McAuley, 2016) are proposed for handling product questions. Although the learning procedures of these models involve

QA collections and reviews, these models assume that the candidate answer set only comes from existing QA pairs containing the correct answers. Such setting is not practical for instant response generation.

We propose a new framework which is able to automatically respond product questions newly posed by users via exploiting question-answer collections and review collections in a coordinated manner. One major subtask in our framework is question-based response review ranking which aims at extracting reviews that are suitable for providing the response to the given question. Learning for response review ranking in this problem setting is challenging since no labeled review sample is available. The existing QA pairs in the corresponding QA collection contains knowledge for response ranking. These QA pairs and the corresponding retrieved review lists can be exploited as distant supervision for question-based response ranking. The relationship between such available QA pairs and reviews can be modeled for facilitating the learning of question-based review ranking.

2 Related Work

Product question answering is an emerging topic. Yu et al. (2012) extract hierarchical structure from the product review collection, and then select sentences from reviews based on the structure. Their model only focuses on review collections. Community Question Answering (CQA) (Zhou et al., 2011) approaches can be adopted to tackle this problem. For example, Zhou et al. (2015) propose to learn continuous word embeddings based on the QA corpus incorporating metadata such as category information, and the learned word embedding can be applied for question retrieval in CQA platform. Chen et al. (2018) encode the question text together with users' social interactions for handling the lexical gap among questions. A random walk based learning method is designed to facilitate the similarities evaluation via the recurrent neural network. One shortcoming of these CQA methods for tackling the task of E-commerce product questions is that they can only make use of the typically small QA collection of the corresponding product, and similar questions associated with other products are not helpful due to different specifications.

QA models (Shen et al., 2017; Yang et al., 2016) try to capture the relation between questions and answers. QA-LSTM (Tan et al., 2016) is developed for question answer matching via bidirectional LSTM with max pooling. The QA model proposed in (Wang and Jiang, 2016) also utilizes LSTM. Learning-to-rank model has also been adopted in some method for question-answer matching, such as in (Surdeanu et al., 2008). Generally existing QA models cannot handle the heterogeneous nature of answer collections and review collections in the problem setting investigated in this paper. McAuley and Yang (2016) propose to exploit product reviews for answer prediction via a Mixture of Expert (MoE) model. This MoE model makes use of a review relevance function and an answer prediction function. One restriction of this model is that it can only be used for answer selection given a candidate answer set. Although the QA collections and review collections are involved in the learning procedures, one assumption is that a candidate answer set containing the correct answers is available for answer selection. Such setting is not practical for instant response generation. In addition, both the above mentioned models only make use of information from reviews, while existing question-answer pairs of the corresponding product are ignored. A chatbot for E-commerce sites known as SuperAgent has been developed (Cui et al., 2017). SuperAgent considers both QA collections and reviews when answering questions. However, there is no mutual coordination for the response generation from different information sources, and the response from SuperAgent cannot be presented as a ranked list of snippets. Moreover, the training procedures of some modules in this system require external knowledge and a large volume of annotated data, including a set of similar question pairs.

3 Our Proposed Framework

3.1 Framework Overview

Our proposed framework exploits QA collections and review collections for instant responses of E-commerce product questions newly posed by users. The output is a ranked list of snippets providing relevant information related to the response of the question. Each snippet can be a sentence from reviews

or an existing QA pair. There are two main components in our framework, namely, response review ranking component and response generation component.

The response review ranking component tackles a major subtask whose aim is to extract response reviews that are suitable for providing the response to the given question. Learning for response review ranking in this problem setting is challenging and it cannot be handled by supervised learning. The reason is that there is no ground truth question-based reviews which can be used for training. The existing QA pairs and the corresponding retrieved review lists contains knowledge for response review ranking. To learn a question-based review ranking model, the relationship between the available QA pairs and reviews needs to be modeled. To this end, we develop a distant supervision paradigm for incorporating the knowledge contained in QA collections into response review ranking. An adapted Learning to Rank (LTR) method is designed in this distant supervision paradigm.

The aim of the response generation component is to generate a list of snippets which serve as the response of a given question newly posed by a user. Both the existing question-answer pairs in the QA collection of the corresponding product and question-based ranked reviews from the review collection of the corresponding product are considered in a coordinated manner. Semantically similar questions are extracted, if exist, via processing the question-answer pairs in the QA collection. The retrieved question-answer pairs are reformulated based on the corresponding review collection, and then integrated into the response review ranking model.

3.2 Response Review Ranking Model

As mentioned above, the aim of response review ranking is to extract reviews that can provide the response to a given question. Consider a E-commerce product t with the corresponding review collection R_t. Given a new question q_0 associated with this product, we wish to obtain the score $S(q_0, r)$ for each review $r \in R_t$. Based on this score, a ranked list of response reviews $L_{r|q_0} = (r_{1|q_0}, r_{2|q_0}, ...)$ can be obtained. A learning model is designed for tackling this subtask. Different from answer selection in CQA, response review ranking cannot be tackled via supervised learning. Suppose we have an existing question q in the question collections, the only available data is the question-answer pair (q, a) while no ground truth question-review pair (q, r) is available for the learning of this component. This setting makes this subtask challenging. Thus, we model the relationship between (q, a) pairs of the corresponding product t and the review set R_t. Such modeling can be regarded as a distant supervision paradigm based on the knowledge in the existing QA pairs for the learning of question-based response review ranking. In this way, we exploit the knowledge in QA pairs in the corresponding QA collection and learn the response review ranking model $p(r|q) \triangleq S(q, r)$, where $S(q, r)$ is the ranking score of the review r given the question q. An adapted Learning to Rank (LTR) method is designed in this distant supervision paradigm, with various types of features are designed.

3.2.1 Distant Learning Paradigm

Training Instance Preparation for Question-based Response Review Ranking. From the QA collections, we can obtain question-answer pairs. Each question-answer pair can provide knowledge about the corresponding product, and neither the question nor the answer alone is adequate for conveying the same semantics as the question-answer pair. We use question-answer pairs as queries for retrieving and ranking reviews, and the top ranked reviews are utilized as the training data for the question-based LTR model.

Given a question-answer pair (q, a), the review collection of the corresponding product is processed. We employ the Positional Language Model (PLM) (Lv and Zhai, 2009) to rank the reviews using (q, a) as the query. Formally, given a review r and a term position k, the PLM of r at this position is:

$$p(w|r, k) = \frac{c'(w, k)}{\sum_{w' \in V} c'(w', k)} \quad (1)$$

where $c'(w,k)$ is the pseudo word count of the word w at the position k:

$$c'(w,k) = \sum_{j=1}^{|r|} c(w,j) f(k,j) \tag{2}$$

and $f(k,j)$ is a weight given to another position. Basically, the closer are the position k and j, the larger value is assigned by $f(k,j)$. In our implementation, we use the Gaussian kernel for $f(k,j)$ as mentioned in (Lv and Zhai, 2009). To obtain an available relevant review ranking list, we design a modified relevance score of PLM:

$$S((q,a),r) = \max_{i\in[1,|r|]} \{S(q,r,i) + S(a,r,i)\} \tag{3}$$

$$= \max_{i\in[1,|r|]} \left\{ -\sum_{w\in V} p(w|q) \log \frac{p(w|q)}{p(w|r,i)} - \sum_{w\in V} p(w|a) \log \frac{p(w|a)}{p(w|r,i)} \right\} \tag{4}$$

We denote the review ranking list based on (q,a) as

$$L_{r|q,a} = (r_{1|q,a}, r_{2|q,a}, ..., r_{n|q,a}) \tag{5}$$

where $r_{i|q,a}$ is the i-th review sentence ranked by the question-answer pair (q,a). The top ranked reviews in this QA-based review ranking list $L_{r|q,a}$ are more relevant to (q,a) and are useful for capturing the knowledge of response review ranking. Thus, we truncate the review ranking list via retaining only the top M ranked reviews as the response reviews.

Since our goal is question-based review ranking, each $L_{r|q,a}$ is integrated with the question q and a training instance is packaged as $(q, L_{r|q,a})$. Here the answer is removed from the training instance even though it participates in the generation of $L_{r|q,a}$. The reason is that there is no available answer for questions newly posed by users during operation or testing. The response review ranking ability of the question-answer pair (q,a) is captured in the response review ranking list $L_{r|q,a}$, and this training instance can be utilized to guide the review ranking based on only the question text. Another issue about the training instance preparation is that different answers have different quality in regard to the question, and the instance based on answers with high quality should have more influence on the learning of the response review ranking model. To achieve this goal, for each question q, we sample training instances from its related response review ranking set $\{L_{r|q,a_1}, L_{r|q,a_2}, ...\}$. The sampling probability of the ranking list $L_{r|q,a_i}$ is the probability of the answer a given the question q, namely $p(a|q)$. We utilize a learning-to-rank model trained with the QA collections to generate $p(a|q)$, and this model makes use of the features as described in Section 3.2.2.

Question-based Learning to Rank. Based on the sampled $L_{r|q,a}$ instances, we utilize the pair-wise learning to rank model to make use of these response review ranking lists to the maximum extent. For each question-review pair, we make use of the features presented in Section 3.2.2. These features are concatenated as a vector denoted as $f^q(r)$. The details about this designed features will be described in Section 3.2.2.

Suppose for two response review r_i and r_j, we obtain the feature vectors $f^q(r_i)$ and $f^q(r_j)$. The probability that the review r_i is ranked higher than the review r_j is formulated as:

$$P^q_{ij} = P(r_i \rhd r_j) = \frac{1}{1 + e^{-\beta[f^q(r_i) - f^q(r_j)]}} \tag{6}$$

To learn a probability model for question-base review ranking, we minimize the cross entropy of the ranking probability generated from the model and the ranking probability from the training instances. Formally, the objective function can be written as:

$$O = -\bar{P_{ij}} \log P_{ij} - (1 - \bar{P_{ij}}) \log(1 - P_{ij}) \tag{7}$$

In our implementation, we utilize the training method with boosting in LambdaMART (Wu et al., 2010) to update the parameters in our response review ranking model.

3.2.2 Feature Design

We describe the features utilized in our LTR models.

- **Features based on Likelihood.** Generation likelihood (Jin et al., 2002) plays an important role in information retrieval. Two related features are designed, namely, the question likelihood given review and the review likelihood given question. For a question q and a review r, the question likelihood can be computed by multiplying the likelihood of query terms w given r, denoted as $p_\lambda(w|r)$. The likelihood is smoothed by Jelinek-Mercer method:

$$p_\lambda(w|r) = (1-\lambda)p(w|r) + \lambda p(w|C_r) \tag{8}$$

  where $p(w|C_r)$ is the probability of the term w in the review collection. Similarly, we can formulate the second feature about the likelihood of the reviews given the question q.

- **Features based on Aspect-based Similarity.** For text data from E-commerce platforms, aspects are an important concept which captures features or attributes about products. Two features related to aspects are designed. The first feature is derived from the aspect discovery model based on Latent Dirichlet Allocation (LDA) (Zhao et al., 2010). Trained with the text collection containing reviews, questions and answers, this model can transform each given text into an aspect representation. Let the representations for a question q and a review r be denoted as $v(q)$ and $v(r)$. We employ the cosine similarity score of these two representations to model the aspect-based similarity between the question and the review. The second feature is based on the Restricted Boltzmann Machine (RBM) (Wang et al., 2015) trained as an aspect modeling. The learned hidden representation from RBM contains aspect information and can be obtained efficiently. Similar to the aspect-based similarity in the first feature, we can also compute a similarity score based on the RBM as the second feature.

- **Features based on Word Embedding** Word embedding techniques map each term to a distributed representation capturing semantics of text. Three features are designed according to different types of word embedding. For the first two features, we adopt a set of pre-trained word embedding, known as Global Vectors for word representation (GloVe)(Pennington et al., 2014). Given a question $q = (t_1^q, t_2^q, ...)$, each term t_i^q in q can be represented as a word embedding denoted as $v(t_i^q)$. Then the question q can be represented as the average of each term representation. The review sentence is also represented in the same way. Then the first feature is obtained by computing the inner product of these two representations reflecting the semantic similarity between the question q and the review r. Another representation of question is to use the largest value in each dimension among all the term vectors as the value of the corresponding dimension in the question representation. Then the second feature can be obtained in a similar manner. Besides the pre-trained word embedding, we also implement an autoencoder with the reconstruction function as the objective function. This autoencoder is trained via the collection of reviews, questions and answers, for capturing the intrinsic elements of text useful for reconstruction. In this way, we obtain a new embedding of questions and reviews, and the corresponding similarity score can be obtained as the third embedding-based feature.

- **Features based on Word Count** Some simple features also contribute to some extent. These features include the word count of a review, the ratio of the review word count and the question word count, and the number of matched words between the question and the review.

3.3 Response Generation

As mentioned above, the aim of the response generation component is to generate a ranked list of snippets treated as the response to the given question. The snippets are extracted from the existing QA collection and the review collection of the corresponding product. To obtain this snippet ranking list, we integrate the question-answer pair result and the review ranking result in a unified ranking model so that the heterogeneous nature between these two collections can be handled in a coordinated manner.

Consider an E-commerce product t with the QA collection Q_t and the review collection R_t. Given a question q_0 associated with this product newly posed by a user, we retrieve semantically similar questions from the QA collection Q_t via searching the question-answer pairs. Specifically, we utilize the Positional Language Model (PLM) (Lv and Zhai, 2009) for question retrieval and obtain a ranked list of retrieved existing questions denoted by $L_{q^*,a^*|q}$. The next step is to utilize the trained LTR model described in Section 3.2 to reformulate each retrieved QA pair $(q_i, a_i) \in L_{q^*,a^*|q}$. The rationale is to narrow the syntactic gap between the QA pairs and the review collection so that QA pairs and reviews can be ranked in a unified model. Suppose that given the new question, we retrieve a similar QA pair (q_1, a_1). We then use (q_1, a_1) as the query to retrieve and rank the reviews in the review collection R_t, and obtain the top ranked reviews $\{r_{1|q_1,a_1}, r_{2|q_1,a_1}, ..., r_{M|q_1,a_1}\}$ in $L_{r|q_1,a_1}$. The reformulation method is designed as follows:

$$r_{q_i,a_i} =< q_i, a_i, \lambda \cdot r_{1|q_i,a_i}, \lambda \cdot r_{2|q_i,a_i}, ... > \tag{9}$$

where $< a, \lambda \cdot b >$ stands for concatenating the bag-of-word model of the two text a and b, and λ is the weight of b for the concatenation. Then we use the trained response review ranking model described in Section 3.2 to rank all the snippets in the set:

$$T = \{r_{q_1,a_1}, r_{q_2,a_2}, ..., r_{q_N,a_N}, \{r\}\} \tag{10}$$

where r_{q_i,a_i} is the pseudo review based on the reformulated QA pairs.

4 Experiments

4.1 Data Collections

We conduct our experiments using two datasets from a real-world E-commerce site, namely *Amazon.com*[1]. The first dataset is from the product category "Cell Phones & Accessories" and the second dataset is from the category "Clothing, Shoes & Jewelry". These two datasets were originally collected by McAuley et al. (McAuley and Yang, 2016; Wan and McAuley, 2016) containing a large number of products, question-answer pairs [2], and reviews [3]. For each product category, we randomly select around 50 questions to form the testing set, and conduct annotations so that these questions can be used for evaluation purpose. The remaining questions are retained as the training set. Regarding the annotation of the testing questions, consider each testing question, we collect the existing QA collection of the corresponding product. Then we examine each QA pair and annotate whether it is semantically similar to a given testing question. Similarly we collect the existing review collection of the corresponding product. We examine each review and annotate whether it can be used as a response for a given testing question. The detailed statistics of the datasets is depicted in Table 1.

Table 1: Statistics of the Datasets

Category	Cell Phones & Accessories	Clothing, Shoes & Jewelry
# of product	10320	2871
# of question	60791	17233
# of review	3447249	5748920
Ave. length of question	12.4	15.2
Ave. length of review sentence	13.9	14.1
# of question for testing	53	55
# of question w/ response reviews	53	55
# of question w/ similar QA Pairs	12	20

[1] https://www.amazon.com

[2] http://jmcauley.ucsd.edu/data/amazon/qa/

[3] http://jmcauley.ucsd.edu/data/amazon/

4.2 Experiment Settings

We conduct sentence segmentation for the reviews. Terms in text such as review sentences, questions, and answer are lemmatized. Stopwords and punctuations are removed. The parameter value of M in Section 3.2 is set to 50. In Jelinek-Mercer smoothing method for LM, the interpolation parameter λ is set to 0.8. For both LDA and RBM in Section 3.2.2, the dimension of hidden layer is set to 10. For pre-trained word embedding, 300-dimension GloVe embeddings are utilized. In the autoencoder for word embedding, the dimension of hidden layer is set to 10.

We implement several baseline and state-of-the-art models to compare with our proposed framework.

- **Language Model (LM)**(Jin et al., 2002). We regard the given new question as the query, and calculate the query likelihood from each review and each question-answer pair via a language model. The list of output snippets are ranked by the question likelihood.

- **Positional Language Model (PLM)** (Lv and Zhai, 2009). Similar to LM, PLM models each document and provides query likelihood considering positional information. The list of output snippets are ranked by the positional question likelihood.

- **QA-LTR** (Surdeanu et al., 2008). QA-LTR is a supervised learning-to-rank model for answer selection, making use of a large volume labeled question-answer pairs. The original QA-LTR model is designed for online QA collections. Hence some features are not suitable. We implement QA-LTR with features replaced by our designed features described in Section 3.2.2. We expand the existing collection of question-answer pairs by including some relevant reviews to form the positive instance training set. Specifically, given a question q in the training set with its answer set A_q, we add similar reviews as candidate answers into A_q. For each $a \in A_q$, we retrieve k review sentences that have the largest similarity to a, denoted as $\{r_1^a, r_2^a, ..., r_k^a\}$. These retrieved reviews are added into the answer set A_q of the question q. We set k to 2 which yields the best performance.

- **QA-LSTM** (Tan et al., 2016). QA-LSTM is one of the state-of-the-art representation learning models for question answering matching. Equipped with bidirectional LSTM and maximum pooling, QA-LSTM has been applied to answer selection. In our experiment, we utilize this model to obtain semantic representation of questions and reviews, and to estimate the matching score given a question q_0 and a review r_0. Similar as in the training of QA-LSTM, the positive instances also include reviews via expanding the existing answer set for each question.

- **Ours.** It denotes our proposed framework.

The evaluation metrics are Normalized Discounted Cumulative Gain (NDCG), Mean Average Precision (MAP), and Mean Reciprocal Rank (MRR). We conduct the t-test for the results to compare the performance improvement over QA-LSTM and QA-LTR, with the significance level being 0.05.

4.3 Product Question Response Performance

Table 2: Response Performance. †, ‡ indicate that Ours is statistical significant at the significance level of 0.05 over QA-LSTM and QA-LTR respectively.

	Cell Phones & Accessories			Clothing, Shoes & Jewelry		
Model	NDCG	MAP	MRR	NDCG	MAP	MRR
LM	0.298	0.137	0.247	0.305	0.146	0.261
PLM	0.307	0.148	0.260	0.321	0.178	0.294
QA-LTR	0.369	0.177	0.403	0.392	0.192	0.320
QA-LSTM	0.397	0.184	0.390	0.407	0.211	0.367
Ours	$0.493^{\dagger\ddagger}$	$0.236^{\dagger\ddagger}$	$0.532^{\dagger\ddagger}$	$0.424^{\ddagger}$	$0.247^{\dagger\ddagger}$	$0.492^{\dagger\ddagger}$

The result is shown in Table 2 for the two product categories. We can observe that our proposed model "Ours" outperforms all the other models. It is statistical significant that our proposed framework

is better than QA-LSTM and QA-LTR. The results indicate that traditional information retrieval method cannot effectively handle the product question response. The reason is that our target is to find the review sentences or existing question-answer pairs that can answer the given question, instead of just retrieving relevant text for the question. The semantic relationship between questions and reviews cannot be modeled via information retrieval models. QA models, namely, QA-LTR and QA-LSTM are not effective as compared to our proposed framework despite the fact that QA models have been previously utilized for review ranking. It reveals that existing QA models have limitation when handling the heterogeneous nature of the review collections and the QA collections. In contrast, the distant supervision paradigm in our framework is more suitable. One main advantage of our proposed paradigm is that it can exploit both information sources in a coordinated manner.

Table 3: Response Performance with different amount of training data.

	Cell Phones & Accessories			Clothing, Shoes & Jewelry		
Model	NDCG	MAP	MRR	NDCG	MAP	MRR
Ours w/ all training data	0.493	0.236	0.532	0.424	0.247	0.492
Ours w/ 1/2 training data	0.483	0.219	0.521	0.421	0.230	0.479
Ours w/ 1/3 training data	0.470	0.201	0.492	0.401	0.215	0.440
Ours w/ 1/4 training data	0.403	0.189	0.453	0.365	0.157	0.398

In addition, we evaluate the performance of our framework under different amount of training data, and the result is presented in Table 3. "Ours w/ 1/n training data" stands for our proposed framework trained with only 1/n of all the training data. We can observe that even when we use a partial amount of the training data, the performance of the response generation does not drop significantly. It demonstrates that our proposed framework can effectively make use of the available training data including QA pairs and QA-based review ranking.

4.4 Case Study

We present a sample case study to gain more insights. Table 4 contains a response snippet list produced from our framework given a new question. The product is a cable and the new question is about how to use the cable for photo transfer to a Mac. The snippets in the response are compposed of review sentences and an existing QA pair. Specifically, the first ranked snippet is a question-answer pair talking about transferring photos from a mobile phone to a Mac. The following snippets come from reviews and they mention that the cable is not for data transferring. Thus, the user can receive informative response extracted from both the review collection and the QA collection associated with the corresponding product.

Table 4: A case study.

Newly posed question
Does anyone know if this cord will work to transfer photos by USB to a Mac? or is it just PC? Thanks!
Response Snippet List
Q: What do I need to get to transfer photos from phone to a Mac if this cable won't work? A: Not quite sure to be honest but maybe if the phone supports like sending emails then you can send your photos through email and download the photos onto your Mac from your email if the cable doesn't work but it should.
It would be nice if it could be used to transfer pictures, but if it can they sure do not make it easy.
Read all descriptions carefully because some of the cables are USB charging cables and do not transfer data!
This is not a data cable.

5 Conclusions

We propose a framework for instant product question response considering two common information sources, namely, existing QA collections and review collections. Our framework exploits both information sources in a coordinated manner. The distant supervision paradigm is adopted for handling a major subtask of question-based response review ranking. Extensive experiments demonstrate the effectiveness and stability of the proposed framework.

References

Zheqian Chen, Chi Zhang, Zhou Zhao, Chengwei Yao, and Deng Cai. 2018. Question retrieval for community-based question answering via heterogeneous social influential network. *Neurocomputing*, 285:117–124.

Lei Cui, Shaohan Huang, Furu Wei, Chuanqi Tan, Chaoqun Duan, and Ming Zhou. 2017. Superagent: A customer service chatbot for e-commerce websites. *Proceedings of ACL 2017, System Demonstrations*, pages 97–102.

Rong Jin, Alex G Hauptmann, and ChengXiang Zhai. 2002. Language model for information retrieval. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 42–48. ACM.

Yuanhua Lv and ChengXiang Zhai. 2009. Positional language models for information retrieval. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 299–306. ACM.

Julian McAuley and Alex Yang. 2016. Addressing complex and subjective product-related queries with customer reviews. In *Proceedings of the 25th International Conference on World Wide Web*, pages 625–635. International World Wide Web Conferences Steering Committee.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Yikang Shen, Wenge Rong, Nan Jiang, Baolin Peng, Jie Tang, and Zhang Xiong. 2017. Word embedding based correlation model for question/answer matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3511–3517.

Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. 2008. Learning to rank answers on large online qa collections. *Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 719–727.

Ming Tan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2016. Improved representation learning for question answer matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 464–473.

Mengting Wan and Julian McAuley. 2016. Modeling ambiguity, subjectivity, and diverging viewpoints in opinion question answering systems. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pages 489–498. IEEE.

Shuohang Wang and Jing Jiang. 2016. Learning natural language inference with lstm. In *Proceedings of The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1442–1451.

Linlin Wang, Kang Liu, Zhu Cao, Jun Zhao, and Gerard de Melo. 2015. Sentiment-aspect extraction based on restricted boltzmann machines. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 616–625.

Qiang Wu, Christopher JC Burges, Krysta M Svore, and Jianfeng Gao. 2010. Adapting boosting for information retrieval measures. *Information Retrieval*, 13(3):254–270.

Liu Yang, Qingyao Ai, Jiafeng Guo, and W Bruce Croft. 2016. anmm: Ranking short answer texts with attention-based neural matching model. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 287–296. ACM.

Jianxing Yu, Zheng-Jun Zha, and Tat-Seng Chua. 2012. Answering opinion questions on products by exploiting hierarchical organization of consumer reviews. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 391–401. Association for Computational Linguistics.

Wayne Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. 2010. Jointly modeling aspects and opinions with a maxent-lda hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 56–65. Association for Computational Linguistics.

Guangyou Zhou, Li Cai, Jun Zhao, and Kang Liu. 2011. Phrase-based translation model for question retrieval in community question answer archives. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 653–662. Association for Computational Linguistics.

Guangyou Zhou, Tingting He, Jun Zhao, and Po Hu. 2015. Learning continuous word embedding with metadata for question retrieval in community question answering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 250–259.