An FAQ Search Method Using a Document Classifier Trained with Automatically Generated Training Data

Takuya Makino(✉), Tomoya Noro(✉), and Tomoya Iwakura(✉)

Fujitsu Laboratories Ltd., Sunnyvale, USA
{makino.takuya,t.noro,iwakura.tomoya}@jp.fujitsu.com

Abstract. We propose an FAQ (Frequently Asked Question) search method that uses classification results of input queries. FAQs aim at covering frequently asked topics and users usually search topics in FAQs with queries represented by bag-of-words or natural language sentences. However, there is a problem that each question in FAQs is not usually sufficient enough to cover variety of queries that have the similar meaning but different surface expressions, such as synonyms, paraphrase and causal relations due to each topic usually consists of a representative question and its answer. As a result, users who cannot find their answers in FAQs ask a call center operator. To consider similarity of meaning among different surface expressions, we use a document classifier that classifies each query into topics of FAQs. A document classifier is trained with not only FAQs but also corresponding histories of operators for covering variety of queries. However, corresponding histories do not include links to FAQs, we use a method for generating training data from the corresponding histories with FAQs. To generate training data correctly, the method takes advantage of a characteristic that many answers in corresponding histories related to FAQs are created by quoting corresponding FAQs. Our method uses a surface similarity between answers in corresponding histories and the answer part of each topic in FAQs for automatically generating training data. Experimental results show that our method outperforms an FAQ search based method using word matching in terms of Mean Reciprocal Rank and Precision@N.

1 Introduction

Call centers are managed to respond their users' question from not only customers of companies but also employees of large companies. To reduce the cost of operators of call centers, FAQs are prepared, which are a set of a question and its answer. Development of FAQs have the following benefits.

1. If users can find answers from FAQs by themselves, the cost of operators is reduced.
2. When users ask call center operators and their answers exist in FAQs, the cost of operators is reduced because operators can answer questions from users by quoting FAQs.

R. Booth and M.-L. Zhang (Eds.): PRICAI 2016, LNAI 9810, pp. 295–305, 2016.
DOI: 10.1007/978-3-319-42911-3_25

Traditional FAQ search methods use a surface similarity between a query and an FAQ for ranking FAQs [2,8]. However, this approach cannot handle queries that have the same meaning but different surface expressions such as synonyms, paraphrase and causal relations. Let us consider the following example.

query: I lost my credit card.
the question part of an FAQ: How do I request a new / replacement card?
the answer part of an FAQ: You can request a replacement card by signing in to the online banking and going to the information tab of your account.

In the above example, only "I" and "card" are matched with the words in the question part of an FAQ, therefore, the score for the answer given by a search based method would not be high.

One of the solutions to this problem is to create knowledge such as a causal relation between "lose" and "replacement". However, the cost of constructing these knowledge should be high because we have to prepare domain specific knowledge in many cases. Some previous works proposed to use a word alignment model for finding semantically similar questions [10,11,13]. They regarded pairs of similar questions or pairs of question and its answer as parallel corpus. Even though their models can learn the probability of a relatedness between two words, their model cannot learn how important a probability of two words for searching correct FAQs.

We propose the use of a document classifier that predicts whether or not a given query corresponds to each FAQ. We need certain amount of training data for training a document classifier that covers variety of surface patterns, however, we usually have only FAQs that are made by grouping past same topic questions and only a representative question of each topic is reserved. To train a document classifier that covers variety of surface patterns, we use training data automatically generated from corresponding histories and FAQs. Corresponding histories do not include links to FAQs, therefore, we take advantage of characteristic that many answers in corresponding histories related to an FAQ are created by quoting the answer part of the FAQ. Our method uses a surface similarity between answers in corresponding histories and the answer part of the topic for automatically generating training data. Then, we train a document classifier with FAQs and the generated training data for predicting corresponding FAQs of a given query. The trained document classifier is used for augmenting features of a learning to rank-based search. Classification results of given queries by the classifier are used features for ranking FAQs.

We evaluate our method with an in-company FAQs and corresponding histories. The experimental results show that our method outperforms a search-based method only uses surface similarities and a word alignment model-based method in terms of MRR, Precision@N.

2 Proposed Method

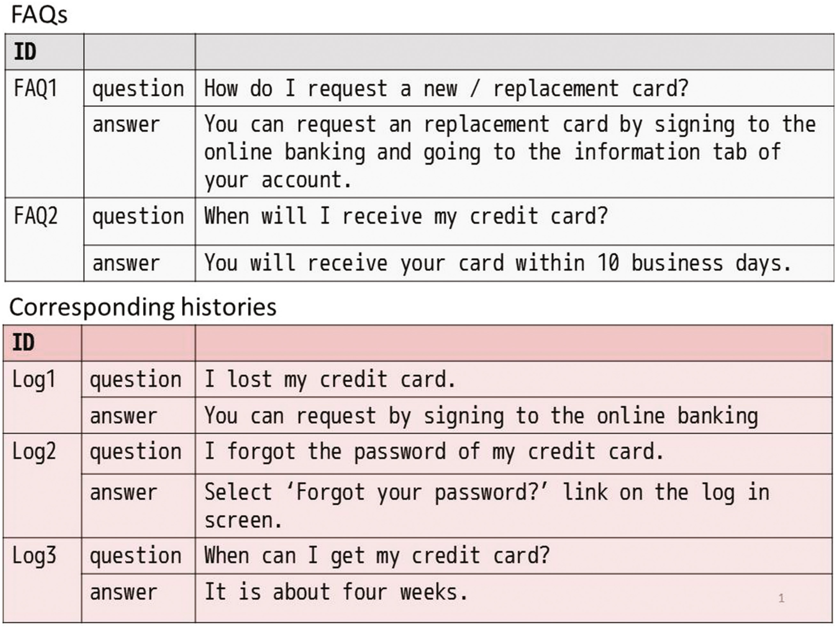
This section describes the problem definition for a learning to rank of this paper and our proposed method. The proposed method consists of the following three steps. At first, our method generates training data automatically. Then, our method trains a document classifier that predicts whether a given question corresponds to each FAQ or not. Finally, our model trains a ranking model to assign higher score for a correct FAQ than an incorrect FAQ for a question.

2.1 Preliminaries

We employ a learning to rank approach for searching FAQs with a given query. The learning objective is to induce a model that assigns higher score to the correct FAQ than the all incorrect FAQs of each given question. The features are not only surface-based ones but also classification results obtained with a document classifier that classifies queries into FAQs. We train a document classifier and a ranking model with FAQs and corresponding histories. FAQs are a set of a question Q and its answer A: $D_1 = \{(Q_1, A_1), ..., (Q_M, A_M)\}$. Corresponding histories are a set of pairs of a user's question I and its answer R: $D_2 = \{(I_1, R_1), ..., (I_N, R_N)\}$. Figure 1 shows examples of FAQs and corresponding histories.

2.2 Generating Training Data Automatically

We have corresponding histories and FAQs, however, answers in corresponding histories do not include links to FAQs. Annotating such data manually is very



FAQs

ID		
FAQ1	question	How do I request a new / replacement card?
	answer	You can request an replacement card by signing to the online banking and going to the information tab of your account.
FAQ2	question	When will I receive my credit card?
	answer	You will receive your card within 10 business days.

Corresponding histories

ID		
Log1	question	I lost my credit card.
	answer	You can request by signing to the online banking
Log2	question	I forgot the password of my credit card.
	answer	Select 'Forgot your password?' link on the log in screen.
Log3	question	When can I get my credit card?
	answer	It is about four weeks.

Fig. 1. Examples of FAQs and corresponding histories.

expensive, therefore, we employ an automatic generation method of training data. To know characteristics of corresponding histories, we first asked operators how they answer for a question. The answers are the following. When an operator receives a question from a user, the operator searches FAQs based on the question. If the operator can find its answer from FAQs, they answer by quoting the answer part of an FAQ. Based on such characteristics, we generate training data automatically by calculating a surface similarity between the answer records in corresponding histories and the answer part of an FAQ.

Our method generates training data automatically by following the previous work [7] that calculates a surface similarity between answer records of corresponding histories. To generate training data automatically, we use FAQs and corresponding histories. FAQs are a set of a question Q and its answer A: $D_1 = \{(Q_1, A_1), ..., (Q_M, A_M)\}$. Corresponding histories are a set of pairs of a user's question I and its answer R: $D_2 = \{(I_1, R_1), ..., (I_N, R_N)\}$.

We use harmonic mean of reciprocal ranks (hrank) that is defined in Eq. 1 as a similarity between the answer records in corresponding histories and the answer part of an FAQ. To handle a large size of corresponding histories, we use the score of a full text search engine Elasticsearch[1] that calculates the tfidf similarity between an answer records in corresponding histories and the answer part of an FAQ. $rank_{A_i}$ indicates the rank of the answer part of an FAQ A_i when searching FAQs by an answer record R_j. $rank_{R_j}$ indicates the rank of an answer record R_j when searching answer records in corresponding histories by the answer part of an FAQ A_i. When $hrank(A_i, R_j)$ is larger than a threshold t for generating training data, our method links an FAQ A_i and its question Q_i to the question I_j of an answer record R_j. Our method generates $D_3 = \{(Q_i, A_i, I_j)|1 \leq m \leq M, 1 \leq n \leq N\}$ as training data.

$$hrank(A_i, R_j) = \frac{1}{2}(\frac{1}{rank_{A_i}} + \frac{1}{rank_{R_j}}) \tag{1}$$

We describe an example of generating training data with FAQs and corresponding histories in Fig. 1. Our method searches corresponding histories with FAQs. Then, our method searches FAQs with corresponding histories. For example, when our method searches answers in corresponding histories with the answer part of FAQ1 in FAQs, the answer of Log1 in corresponding histories is ranked at the first place. Next, our method searches answer parts of FAQs with the answers in corresponding histories. Here, the answer part of FAQ1 in FAQs is ranked at the first place. In this situation, the hrank of FAQ1 and Log1 takes 1 and our method appends FAQ1 and the question of Log1 ((“How do I request a new / replacement card?”, “You can request an replacement card by signing to online banking and going to information tab for your account.”), “I lost my credit card.”) to training data D_3.

[1] https://www.elastic.co/jp/.

2.3 Training a Document Classifier

We train a document classifier with the automatically generated training data in addition to FAQs. Since the number of FAQs is more than two, we employ a multi class classification method based a one-versus-the-rest method. A binary document classifier for each FAQ is trained with the questions that are linked with the FAQ as positive examples, the questions that are linked with the other FAQs as negative examples. Question parts of FAQs are also used for training. When our model trains a binary document classifier for an FAQ (Q_i, A_i), questions that are linked to (Q_i, A_i) in D_3 and Q_i are used as positive examples, and questions that are linked to (Q_k, A_k) in D_3 $(1 \leq k \leq M \text{and} k \neq i)$ is used as negative examples.

For training binary classifiers, we use Adaptive Regularization of Weight Vectors [5] with the following features. We use unigrams of content words in base forms, word bigrams, pairs of content words in base forms that has a dependency relation as features. Except for word bigrams, we convert a word to a base form.

For example, when training a binary classifier for the FAQ "How can I request a replacement card" and positive example is "I lost my credit card.". Our method extracts the following text representations as features from a given question:

Unigrams of content words in base forms: "lose", "credit" and "card"
Bigrams of words: "(I, lost)", "(lost, my)", ..., "(credit, card)"
A pair of content words that have a dependency relation: "lose→card" and "lose→credit"

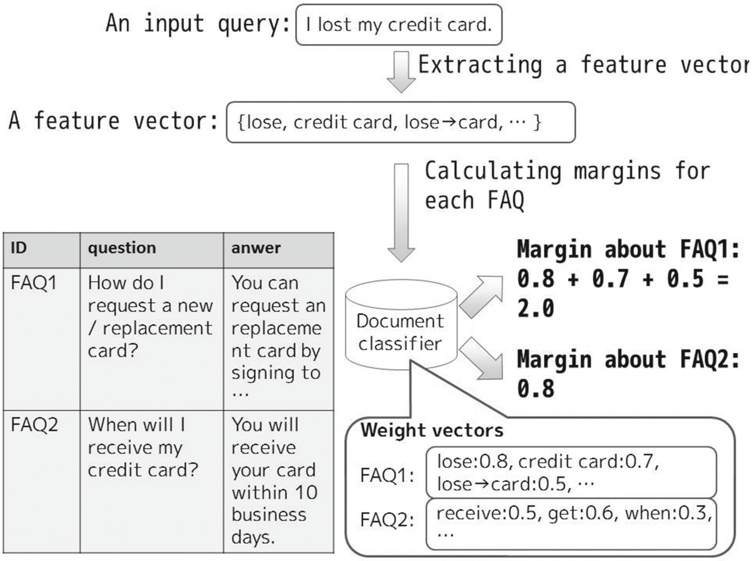
2.4 Learning a Ranking Model

To train a ranking model, we use the automatically generated training data in addition FAQs as in the document classifier training. Our approach is based on a pairwise learning algorithm that learns a ranker to assign higher score for a correct FAQ than an incorrect FAQ.

We show our learning to rank procedure in Algorithm 1. For a question in the automatically generated training data, GetFeatVec$(\hat{Q}, \hat{A}, I)$ extracts a feature vector from a question and a correct FAQ. GetRndFalsePair(I, D_1) randomly samples an incorrect FAQ and GetFeatVec(Q_k, A_k, I) extracts a feature vector from a question and an incorrect FAQ. UpdateWeight$(\mathbf{w_r}, \mathbf{x})$ updates a weight vector with difference of these two feature vectors. ϕ_r indicates a feature vector extracted from input question I, both of the question part Q and the answer part A of an FAQ. We use Adaptive Regularization of Weight Vectors [5] for updating weight vector and set K to 10.

A GetFeatureVec(Q_i, A_i, I) function in Algorithm 1 extracts the following features:

- **cos-q, cos-a**: A cosine similarity between a given question and the question part of an FAQ and a cosine similarity between a given question and the answer part of an FAQ


An input query:
I lost my credit card.
Extracting a feature vector
A feature vector:
{lose, credit card, lose→card, ⋯ }
Calculating margins for each FAQ
Document classifier
Margin about FAQ1: 0.8 + 0.7 + 0.5 = 2.0
Margin about FAQ2: 0.8
Weight vectors
FAQ1:
lose:0.8, credit card:0.7, lose→card:0.5, ⋯
FAQ2:
receive:0.5, get:0.6, when:0.3, ⋯

ID	question	anwer
FAQ1	How do I request a new / replacement card?	You can request an replaceme nt card by signing to ⋯
FAQ2	When will I receive my credit card?	You will receive your card within 10 business days.

Fig. 2. The example of calculating margins for each FAQ by using a document classifier

Algorithm 1. pairwise learning to rank

```
$\mathbf{w_r} \leftarrow \mathbf{0}$
for $(\hat{Q}, \hat{A}, I) \in D_3$ do
    $\mathbf{x}_p \leftarrow \mathrm{GetFeatVec}(\hat{Q}, \hat{A}, I)$
    for k do1...K
        $(Q_k, A_k, I) \leftarrow \mathrm{GetRndFalsePair}(I, D_1)$
        $\mathbf{x}_n \leftarrow \mathrm{GetFeatVec}(Q_k, A_k, I)$
        $\mathbf{x} \leftarrow \mathbf{x}_p - \mathbf{x}_n$
        $\mathbf{w_r} \leftarrow \mathrm{UpdateWeight}(\mathbf{w_r}, \mathbf{x})$
    end for
end for
```

- **dep**: The number of pairs of words each of which has a dependency relation.
- **np**: The number of noun phrases that each of which occurs in both a given question and the question part of an FAQ divided by the number of noun phrases that occur either a given question and the question part of an FAQ.
- **syn**: Whether the same synset occurs in the both of question and question part of FAQ. A synset is a group of word senses that has similar meaning in WordNet.
- **faq-cat**: Whether the category of an FAQ exists in predicted top-5 categories of a given question. Since an FAQ has categories, we used category information of an FAQ. We train a document classifier that predicts the category for a given question because an FAQ has categories but a given question does not have

categories. We train a category classifier by using bag-of-words as features and employs Adaptive Regularization of Weight Vectors [5] for learning.
– **faq-scorer**: We use the margin of a binary classifier for an FAQ (Q_i, A_i) with a sigmoid fitting as a feature.

In a training phase of a ranker, our model uses the difference of feature vectors between true FAQ and false FAQ for learning $\mathbf{w}_r$. In a test phase a ranker, our model extracts a feature vector for each FAQ, and ranks FAQs by sorting scores of FAQs assigned by the trained ranker.

3 Experiments

3.1 Experimental Setup

We used an in-company FAQs and corresponding histories for experiments. For creating test data, we annotated randomly sampled 286 questions from corresponding histories with their correct FAQs.

For generating training data, we set the threshold for automatically generating training data to 0.6. If the answer part of an FAQ is too short, linked questions in corresponding histories may include noise. Therefore, we discarded generated training data for FAQs those number of characters in answer part is less than 10. Our method generated $27,040$ pairs of questions and FAQs. We removed questions that exists in the both of the training data and the test data for training our model.

We used a Japanese morphological analyzer MeCab[2] for word segmentation and a Japanese dependency parser CaboCha[3] for dependency parsing. We used Mean Reciprocal Rank (MRR) and Precision@N (P@N) as evaluation metrics. MRR is the average of reciprocal ranks of correct FAQs and it is going to 1 when correct FAQs ranked higher than incorrect FAQs. P@N is the ratio of a correct FAQ that are ranked higher than Nth and it is going to 1 when correct FAQs are ranked higher than Nth.

We compare our proposed method with baselines that are a full text search engine Elasticsearch and a word alignment model-based one [7]. A word alignment model-based model is formalized as follows:

$$P(Q|I) = \prod_{w \in Q} P(w|I), \tag{2}$$

where $P(w|I)$ is calculated as

$$P(w|I) = (1-\lambda) \sum_{t \in Q} (P_{tr}(w|t) P_{ml}(t|I)) + \lambda P_{ml}(w|C). \tag{3}$$

We estimated $P_{tr}(w|t)$ with the automatically generated training data by using GIZA++[4]. Following Jeon et al. [7], we set $P_{tr}(w|w) = 1$. We set λ that maximizes MRR of test data.

[2] https://taku910.github.io/mecab/.
[3] https://taku910.github.io/cabocha/.
[4] http://www.statmt.org/moses/giza/GIZA++.html.

3.2 Experimental Results

Evaluation of Automatically Generated Training Data: We randomly sampled 50 pairs from automatically generated training data and evaluated those pairs by a human annotator. We show the evaluation result in Table 1.

Almost half of pairs are correct. When the answer part of an FAQ is short, paired questions are noisy.

Table 1. Evaluation of automatically generated pairs

label	number
true	24
false	26

Evaluation of an FAQ Category Classifier: The in-company FAQ used in our experiment has hierarchical categories. We used categories at depth two and number of categories is 107. We conducted 10-fold cross validation using FAQ. Since FAQ has categories which is assigned in advance. We expected that FAQ category classifier can predict that category-level similarity between question and FAQ. We used Adaptive Weight Regularization of Weight Vectors [5] for updating a parameter vector. We show the results of FAQ category classification in Table 2.

Table 2. P@N for FAQ category classification

	P@N
P@1	0.758
P@2	0.839
P@3	0.872
P@4	0.889
P@5	0.898

Results of FAQ Ranking: We show ranking results in Table 3. For full text search, we use search queries consist of content words which are joined with OR. Our proposed method outperformed full text search and word alignment-based model in terms of MRR, P@1, P@5, P@10.

We conducted ablation tests and Table 4 shows the evaluation results. We can see that the contribution of faq-scorer is the largest in our feature set.

Figure 3 shows an MRR learning curve of our proposed method. To plot MRR, we selected 1,000 training data and incrementally learned our model at each step. MRR of our proposed method is improved by increasing the number of training data.

Table 3. Comparison with baselineValues with † significantly differ from our proposed method. We conducted paired t-test in terms of MRR, P@1, P@5, P@10.

method	MRR	P@1	P@5	P@10
Proposed Method	0.478	0.367	0.605	0.727
word alignment-based model	0.315†	0.238†	0.402†	0.476†
full text search	0.276†	0.174†	0.388†	0.483†

Table 4. Ablation tests

method	MRR	P@1	P@5	P@10
Proposed method	0.478	0.367	0.605	0.727
w/o syn	0.478	0.367	0.601	0.727
w/o dep	0.478	0.363	0.612	0.731
w/o np	0.476	0.360	0.605	0.717
w/o faq cat	0.469	0.357	0.598	0.710
w/o cos-{q,a}	0.397	0.311	0.486	0.605
w/o faq scorer	0.346	0.220	0.486	0.601

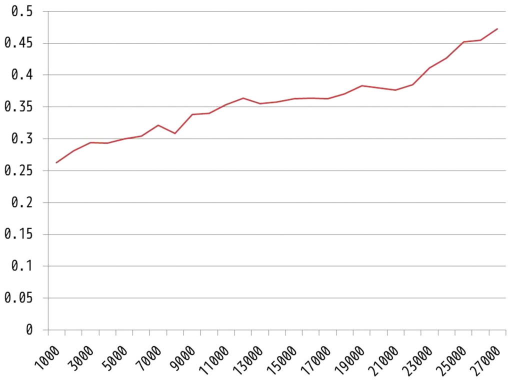


Fig. 3. Learning curve of MRR of our proposed method

Analysis of FAQ Scorer: Table 5 shows features that have larger weight for FAQ “How can I request a new / replacement card?”. For example, this FAQ can correspond to a question that has “lose→card”, “magnetic failure” or “wallet”, which are not included in the original question of the FAQ.

Error Analysis: Main reason of failure of our method is that some FAQs do not have paired questions. If the answer part of an FAQ is similar with the answer part of the other FAQ, our method tends to fail because the questions that are linked automatically to those FAQs have similar contents.

Table 5. Features that have positive weight

Feature name	Feature
dependency	lose→card
noun phrase	replacement application
word bigram	magnetic failure
dependency	card→stole
word unigram	lose
word unigram	wallet

4 Related Works

For searching FAQs or a community QA site such as Yahoo! Answers, some previous research used WordNet or Wikipedia [2,14] for using synonym dictionaries. Since WordNet and Wikipedia do not cover the topic of FAQs, it's difficult to use knowledge such as synonyms.

There are some researches propose the use of IBM models [1] that learn a probability of two words with a statistical machine translation algorithm [7,10,11,13]. These models learn an alignment probability of two words. It is important to align two words in machine translation, however, an alignment probability does not indicate an importance for finding correct FAQs.

Cao et al. [3,4] proposed to use category information for language model. Their models calculate a probability that a given question belongs to the category of a candidate question and assign it to an alignment probability. Their methods are similar to our model in terms of using a document classifier, but our document classifier predicts directly whether a given question corresponds to each FAQs or not directly.

Ko et al. [9], Surdeanu et al. [12] and Higashinaka and Isozaki [6] used learning to rank for question answering. Their models are similar to our model in terms of learning to rank but features of their model are based on a surface similarity, an alignment probability, a dictionary based similarity and a query log based similarity. Our model uses the output of a document classifier that predicts whether a given question corresponds to FAQs or not.

5 Conclusion

We proposed an FAQ search method that uses classification results of input queries. For training a document classifier, our method generates training data automatically. By utilizing a document classifier that predicts whether a given question corresponds to each FAQ or not, our method outperformed baselines, which are a full text search and a word alignment model-based one. In the future work, we need to improve the generating method of training data.

References

1. Brown, P.F., Pietra, V.J.D., Pietra, S.A.D., Mercer, R.L.: The mathematics of statistical machine translation: parameter estimation. Comput. Linguist. (1993)
2. Burke, R., Hammond, K., Kulyukin, V., Lytinen, S., Tomuro, N., Schoenberg, S.: Natural language processing in the FAQ finder system: results and prospects. In: Working Notes from AAAI Spring Symposium on NLP on the WWW (1997)
3. Cao, X., Cong, G., Cui, B., Jensen, C.S.: A generalized framework of exploring category information for question retrieval in community question answer archives. In: Proceedings of the WWW (2010)
4. Cao, X., Cong, G., Cui, B., Jensen, C.S., Zhang, C.: The use of categorization information in language models for question retrieval. In: Proceedings of CIKM (2009)
5. Crammer, K., Kulesza, A., Dredze, M.: Adaptive regularization of weight vectors. In: Proceedings of NIPS (2010)
6. Higashinaka, R., Isozaki, H.: Corpus-based question answering for why-questions. In: Proceedings of IJCNLP (2008)
7. Jeon, J., Croft, W.B., Lee, J.H.: Finding similar questions in large question and answer archives. In: Proceedings of CIKM (2005)
8. Jijkoun, V., de Rijke, M.: Retrieving answers from frequently asked questions pages on the web. In: Proceedings of CIKM (2005)
9. Ko, J., Mitamura, T., Nyberg, E.: Language-independent probabilistic answer ranking for question answering. In: Proceedings of ACL (2007)
10. Riezler, S., Vasserman, A., Tsochantaridis, I., Mittal, V., Liu, Y.: Statistical machine translation for query expansion in answer retrieval. In: Proceedings of ACL (2007)
11. Soricut, R., Brill, E.: Automatic question answering using the web: beyond the factoid. Inf. Retr. **9**, 191–206 (2006)
12. Surdeanu, M., Ciaramita, M., Zaragoza, H.: Learning to rank answers on large online QA collections. In: Proceedings of ACL (2008)
13. Xue, X., Jeon, J., Croft, W.B.: Retrieval models for question and answer archives. In: Proceedings of SIGIR (2008)
14. Zhou, G., Liu, Y., Liu, F., Zeng, D., Zhao, J.: Improving question retrieval in community question answering using world knowledge. In: Proceedings of IJCAI (2013)