Question Utility: A Novel Static Ranking of Question Search

Young-In Song
Korea University
Seoul, Korea
song@nlp.korea.ac.kr

Chin-Yew Lin, Yunbo Cao
Microsoft Research Asia
Beijing, China
{cyl, yunbo.cao}@microsoft.com

Hae-Chang Rim
Korea University
Seoul, Korea
rim@nlp.korea.ac.kr

Abstract

In this paper, we propose a notion of 'question utility' for studying usefulness of questions and show how question utility can be integrated into question search as static ranking. To measure question utility, we examine three methods: (a) a method of employing the language model to estimate the probability that a question is generated from a question collection and then using the probability as question utility; (b) a method of using the LexRank algorithm to evaluate centrality of questions and then using the centrality as question utility; and (c) the combination of (a) and (b). To use question utility in question search, we employ a log linear model for combining relevance score in question search and utility score regarding question utility. Our experimental results with the questions about 'travel' from Yahoo! Answers show that question utility can be effective in boosting up ranks of generally useful questions.

Introduction

Over the past few years, community-based Q&A (cQA) services (e.g. Yahoo! Answers[1], Live QnA[2]), as one of important online venues where people sharing their knowledge, have accumulated vast archives of questions and their answers. It is obvious that such large archives of questions and answers are of great value when re-used as public knowledge base. However, due to the lack of editorial control, questions in the archives vary greatly in their usefulness, which complicates the reuse of the knowledge base. At cQA services, users can issue any questions, even ones clearly useless to other people.

In this paper, we address the issue raised above within the setting of question search (Jeon, Croft, and Lee 2005; Riezler et al. 2007). Like web search (Richardson, Prakash, and Brill 2006), it is required that a question search system not only returns relevant questions to a query, but also order the returned questions based on the 'usefulness of questions'. That is, a question search system should be able to separate useful questions from useless ones. For example, suppose that a question search system only retrieves two questions with respect to the query 'best of Boston':

- *Q1: What is the best hospital to work at in Boston?*
- *Q2: Where is the best restaurant in Boston (and metro area)?*

If only relevance between a query and targeted questions is considered, both questions ask "something best in Boston" and thus can be considered relevant to the user's query. To general people, however, they seem to be quite different in usefulness. Compared to the question $Q2$, the question $Q1$ can be useful only for a restricted group of people. Thus, if we consider usefulness, $Q2$ should be ordered (or ranked) before $Q1$.

To date, most previous work (Burke et al. 1997; Sneiders 2002; Jeon, Croft, and Lee 2005; Riezler et al. 2007) on question search only focuses on how to model relevance between query and targeted questions while ignoring *usefulness* of targeted questions. We note that usefulness of targeted questions only depends on the targeted questions themselves, not related to queries. Thus, as an analog of *static ranking of web search*, we refer to usefulness as *static ranking of question search*. However, we cannot make use of the algorithms (e.g., PageRank, Brin *et al.* 1998) on the basis of link analysis because there don't exist any meaningful links among questions.

In this paper, we propose a notion of '*question utility'* for studying *usefulness of questions*. 'Question utility' is defined as the possibility that a question would be repeated by other people. Our intuition here is quite straightforward: The more times is a question asked (repeated), the more useful the question is. To model 'question utility', we examine two different methods: the language modeling method and the LexRank method.

We also propose to use 'question utility' as *static ranking of question search*. We build a log linear model for question search, which naturally combines *utility* score and *relevance* score in a language modeling framework.

We empirically evaluate the use of *question utility* in question search with a data set consisting of 100 short keyword-based queries and 310,000 questions about 'travel'. The queries are from the query log of Live Search[3] and the questions are from Yahoo! Answers. Experimental results show

[1]http://answers.yahoo.com

[2]http://qna.live.com

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

that *question utility* can significantly boost the performance of question search when used to re-order the search results.

Related Work

In this section, we will briefly review the work related to question search and the work related to using static ranking for (web) search.

Question Search

Most previous work on question search focuses on how to model relevance between a query and targeted questions. The research of question search is first conducted using FAQ data (Burke et al. 1997; Sneiders 2002). Recently, the research has been further extended to the cQA data. For example, Jeon et al. (2005) and Riezler et al. (2007) examined translation-based methods for fixing the lexical chasm and showed its effectiveness in better modeling relevance between a query and targeted questions. However, none of the existing researches addresses usefulness of questions and its influence on question search. For the first time, we propose a notion of 'question utility' for characterizing the usefulness of questions and then try to use it as static ranking of question search.

Static Ranking for Search

Within the area of web search, there exist a large number of researches devoted to modeling or employing static ranking which is referred to as 'static ranking of web search'. PageRank, which is on the basis of link structure (given by hyperlinks) analysis (Brin and Page 1998; Kleinberg 1998), is the most representative algorithm of such research. Due to the absence of link structures among questions, however, the algorithm cannot be applied to question search.

Several alternative ways have been proposed for static ranking, which do not assume the existence of explicit structural features. For instance, Kurland and Lee (2005) suggested to induce implicit link structure based on similarity between texts for static ranking of document retrieval. Zhou and Croft (2005) proposed document quality models using contents based features such as the information noise ratio. Erkan and Radev (2004) proposed the LexRank for automatic document summarization, which is also based on text similarity. In this paper, we investigate the use of the LexRank for evaluating question utility.

The work most related to ours is the research using quality of answers as static ranking of question search (Jeon et al. 2006). It uses quality of answers as the prior of questions for question search. While, question utility is a kind of prior that is on the basis of questions themselves. In question search, a highly-ranked question should be a useful one with high-quality answers. Thus, quality of answers and question utility are two orthogonal characteristics as static ranking of questions. In this paper, we will focus on studying question utility as static ranking of question search and leave the combination of both as our future work.

Utility of Questions

Recall that questions archived from online knowledge bases vary in their usefulness. To better make use of the questions, we need to figure out how to characterize the *usefulness of questions*, which is referred to as *'question utility'* in this paper.

The judgment on usefulness of questions is very subjective. That is, the usefulness of a specific question can vary from person to person. Thus, the notion of *'question utility'* that we define is not based on the judgment from an individual user but based on that from a mass of users. Only the questions that a mass of users considers useful can be of high *question utility*.

Usually, *a user asks a question just because she or he is interested in the (potentially useful) answers to the question.* Thus, if a question is frequently asked, the question should be considered useful (or of *high utility*) by a mass of users.

On the basis of the discussion above, we define question utility as **the likelihood that a question is repeatedly asked by people**.

Language Model for Evaluating Question Utility

We regard the task of evaluating *question utility* of a question Q as one of estimating $p(Q)$, the probability that a question Q can be generated.

One simple method for estimating the probability $p(Q)$ is just to count all the occurrences of the question within a certain question collection C and then normalize the counts into probability by the total number of questions in C. However, because a natural language question can be expressed in various surface forms and it is not easy to normalize all questions semantically equivalent to each other into a single surface form, the simple method cannot be feasible.

The language modeling approach (Manning and Schütze 1999) can be an effective framework to solve this problem. The language model is aimed to estimate the probability of a natural language text being generated and its effectiveness has been proven in many natural language processing and speech recognition related areas. Also, by its ability of decomposing a whole context into small pieces, the language model can avoid counting occurrences of questions directly.

Formally, we use n-gram language model to estimate the probability $p(Q)$ as follows:

$$p(Q) = p(q_1, q_2, \dots, q_m) \approx \prod_{i=1}^{m} p(q_i | q_{i-n+1}^{i-1}) \qquad (1)$$

where q_i denotes the ith word in Q, and q_{i-n+1}^{i-1} means a n-gram word sequence from a word q_{i-n+1} to q_{i-1}. If n is set to 1, the equation (1) is about unigram language model.

To alleviate the data sparseness problem, we smooth the n-gram probability $p(q_i | q_{i-n+1}^{i-1})$ with the Katz back-off method (Katz 1997) which is widely used and shown good performance in speech recognition.

It is known that the language modeling approach is biased such that short texts (in our case, short questions) are of high probabilities. However, we can easily find the long questions repeated many times while occurring in various surface forms. For example, *"What is the must-see things in Paris?"* and *"What should I see in Paris?"*

To avoid the bias that we don't expect to have, we introduce a length normalization method to regularize the raw probabilities:

$$p_{norm}(Q) \propto \exp\left[\frac{\log p(Q)}{\log(m+\alpha)}\right] \quad (2)$$

where α is a smoothing parameter in case that m equals 1. In our experiments, it is set 0.1.

LexRank Method for Evaluating Question Utility

One alternative way to evaluate *question utility* is to identify the most central questions in a given question collection based on the lexical centrality: If a topic is very useful to people, there can be many lexically similar questions related to the topic in the question collection. Among those similar questions, the most central questions can be regarded as the most representative (or useful) ones regarding the topic.

To measure such lexical centrality of questions, we use the LexRank method which is proposed for document summarization by Erkan and Radev (2004). The LexRank estimates the centrality of a sentence (in our case, a question) in a manner similar to the PageRank (Brin and Page 1998) algorithm:

First, from a question collection, lexically similar questions are selected by examining cosign similarity between questions, and on the basis of it, an undirected graph of questions is constructed. In the graph, each node represents a question and two nodes are edged such that similarity between them exceeds a certain threshold value.

Second, for a question Q, the centrality $c(Q)$ is calculated based on the Random Walk algorithm with the following weighting scheme:

$$c_i(Q) = \frac{d}{N} + (1-d)\sum_{v \in adj[Q]} \frac{c_{i-1}(v)}{deg(v)} \quad (3)$$

where N is the total number of nodes in the graph, d is a dampening factor, $adj[Q]$ is the set of nodes adjacent to Q, and $deg(v)$ denotes the degree of node v (the number of its adjacent nodes). For the detailed description, see (Erkan and Radev 2004).

Also, we can combine the language modeling approach described in the previous section with the LexRank method by using utility score from a language model as an initial value in the LexRank weighting scheme (Otterbacher, Erkan, and Radev 2005):

$$c_i(Q) = d \cdot p(Q) + (1-d)\sum_{v \in adj[Q]} \frac{c_{i-1}(v)}{deg(v)} \quad (4)$$

where $p(Q)$ is the likelihood of the question Q estimated by language model. We will investigate the effectiveness of the both method in our experiments.

Using the Utility as Static Ranking of Search

In this paper, we confine ourselves to use *question utility* as *static ranking of question search*. The use of *question utility*, however, can be extended to other applications of reusing questions, too.

In terms of question retrieval, the query likelihood retrieval model is defined as the probabilistic function of generating a user's query Q' from the question language model Q as follows:

$$p(Q|Q') = \frac{p(Q'|Q)p(Q)}{p(Q')} \propto p(Q'|Q)p(Q) \quad (5)$$

where $p(Q')$, the likelihood of a query, does not affect the ranking of questions so it can be ignored by the rank preserving principle (Jones, Walker, and Robertson 2000).

Then, the generation probability $p(Q'|Q)$ is decomposed into a unigram model by using zero order Markov assumption:

$$p(Q|Q') \propto p(Q)\prod_{w \in Q'} p(w|Q) \quad (6)$$

In this equation, $p(Q)$ is the prior probability of question Q reflecting a static rank of the question, independent on a query Q'. Since our utility of a question is defined as the likelihood of a question regardless of a specific query, it can be naturally integrated in this retrieval framework. Therefore, in our approach, we use a value calculated by the proposed methods for a question Q as a value of the probabilistic term $p(Q)$ in the equation (6).

To add flexibility to control the importance of each factor in the retrieval model, we modify the equation (6) into a log linear form:

$$p(Q|Q') \propto \frac{1}{Z(\lambda_1,\lambda_2)}\left[\lambda_1 \log p(Q) + \lambda_2 \sum_{w \in Q'} \log p(w|Q)\right] \quad (7)$$

where λ_1 and λ_2 are interpolation parameters, and $Z(\lambda_1,\lambda_2)$ is a normalization factor. Because the normalization factor $Z(\lambda_1,\lambda_2)$ also does not affect the ranks of question, we can modify the equation (7) by removing $Z(\lambda_1,\lambda_2)$ and introducing a new constant α ($\alpha = \lambda_1/\lambda_2$):

$$p(Q|Q') \propto \alpha \cdot \log p(Q) + \sum_{w \in Q'} \log p(w|Q) \quad (8)$$

To estimate the unigram probability $p(w|Q)$, we use the linear interpolated smoothing (Zhai and Lafferty 2001):

$$p(w|Q) = \lambda_d \cdot p_{mle}(w|Q) + (1-\lambda_d) \cdot p_{mle}(w|C) \quad (9)$$

where $p_{mle}(\cdot)$ indicates a probability estimated using maximum likelihood estimator, C denotes a question collection and λ_d is the smoothing parameter. In our experiments, the optimal value for λ_d has been empirically determined by an exhausted search of the parameter space.

Empirical Evaluations

Our empirical evaluations consist of two experiments. One is to evaluate the proposed approach to assessing question utility. The other is to evaluate the use of question utility as static ranking of question search.

Table 1: The Statistics on the Ground Truth 'SET-B' on Question Utility

	Los Angles	Paris	Beijing	Seoul	Tokyo
#related	1,944	2,217	516	276	817
#relevant	194	285	96	46	158

Experiment Setup

Source Data. We made use of the questions obtained from Yahoo! Answers for the evaluation. More specifically, we utilized the *resolved* questions about 'travel' at Yahoo! Answers. The questions include about 310,000 items. Each resolved question consists of three fields: 'title' presenting major information of a question, 'description' presenting additional detailed context of a question, and 'answer'. In our experiments, questions refer to texts in the 'title' field. We refer to the data set as 'SRC-DAT'.

Question Utility Evaluation Data. In order to evaluate the performance of our language model based approach to assessing question utility, we selected five city names (namely Los Angles, Paris, Beijing, Seoul, and Tokyo) as topics and then built a ground truth against the topics with the following steps.

First, 10 persons were asked to independently compose questions regarding the five topics. Specifically, for each topic, each person was requested to provide 10 questions that he (or she) most probably asks when planning to visit the city (topic). As a result, 100 questions were collected with respect to each topic. We refer to the data set as 'SET-A'. Then, an assessor was asked to manually select questions from 'SRC-DAT' to form the ground truth (denoted as 'SET-B') by observing the data set 'SET-A'. A question was selected provided that the question can be used as the reference for answering certain question from 'SET-A'. As the ground truth, we made use of 'SET-B' for evaluating our approach to question utility because none of questions in 'SET-A' can be found in 'SRC-DAT'.

Our idea to this evaluation method is simple: If a system can predict general usefulness of a question more precisely, there would be more chance that questions ranked highly by a system can cover many questions regarded as useful one by individual people.

Table 1 provides the statistics on the ground truth 'SET-B'. '#related' refers to the number of questions regarding the corresponding city in the data set 'SRC-DAT' and '#relevant' refers to the number of questions in 'SET-B'. Table 2 show the examples from 'SET-A' and 'SET-B' respectively.

Question Search Evaluation Data. In order to evaluate the use of question utility as static ranking of question search, we randomly selected 100 queries from the query log of a commercial web search engine and then built a ground truth on question search.

A query is selected only provided that it contains more than two words and is related to the 'travel' domain. The average length of the 100 queries is 3.5 words (after removing the stopwords, it is 2.7 words). Table 3 shows several examples of queries used in our experiments. 'Frequency' is the number of occurrences of the corresponding query in the query log.

Table 2: The Example Questions in the Ground Truth on Question Utility

Data Set	Example Questions
SET-A	Where are the good and inexpensive places to stay in Paris (nearby downtown)?
SET-B	1. I am visiting Paris this winter. Can anyone suggest affordable hotel near to center? 2. Can anyone recommend inexpensive place to stay in Paris? 3. Where is a good and cheap place to stay and eat in Paris?

Table 3: Some Example Queries in the Ground Truth on Question Search

Query	Frequency
houston texas restaurant	831
airfare to germany	669
pregnancy and travel	666
great wall of china	62,097
what to pack for a cruise	687

For each query, we use the language modeling approach (Zhai and Lafferty 2001) to search for related questions from the data set 'SRC-DAT'. Then, two assessors were asked to manually determine the relevance of the top 200 related questions independently. For the questions that two assessors did not agree on, one additional annotator was asked to provide judgment as the final annotation.

Evaluation Metrics. We have conducted two experiments as ranking tasks. The experiment evaluating our approach to assessing question utility is considered as a static ranking task. The experiment evaluating the use of question utility for search is a dynamic (search) ranking task. Thus, we made use of the standard IR evaluation metrics such as Mean Average Precision (MAP), R-Precision (R-prec), and P@N.

Other Configurations. In the experiments, we stemmed words with the Porter stemmer and removed a few stopwords such as 'a', 'an', and 'the' at both indexing and runtime searching stages. As for the training of language models, we use the CMU-Cambridge Statistical Language Modeling Toolkit[4]. In the training, stemming and stopword removal were also conducted.

Results

Assessing Question Utility by Language Model. For our language modeling method, we tried two kinds of variances: (a) unigram model vs. trigram model; (b) the length normalization (as given by equation 2) or none.

To our observation, long questions often ask for too-personalized or un-popular things, which cannot be considered useful by a mass of people. Thus, we consider that the method of ranking questions (within each topic) by inverse

[4]http://svr-www.eng.cam.ac.uk/ prc14/toolkit.html

Table 4: Assessing Question Utility
('+NORM indicates that the length normalization is used and a number in each cell indicate MAP score for each method.)

Method	LA	Paris	Beijing
Inverse Length	0.098	0.092	0.164
Unigram	0.155	0.184	0.297
Unigram + Norm	0.244	0.197	0.334
Trigram	0.198	0.230	0.370
Trigram + Norm	**0.272**	**0.294**	**0.400**

	Seoul	Tokyo	Average
Inverse Length	0.120	0.226	0.140
Unigram	0.307	0.233	0.235
Unigram + Norm	0.387	**0.354**	0.303
Trigram	0.350	0.264	0.282
Trigram + Norm	**0.393**	0.348	**0.341**

Table 5: Comparison of Top 3 Ranked Results for L.A. topic

Inverse Length	Trigram	Trigram+Norm
1. Downtown LA?	1. Go to LA?	1. **What are some fun things to do in LA?**
2. Go to LA?	2. Live in L.A.?	2. **Where is the best place to live in LA?**
3. NYC vs. L.A?	3. **Hotel in LA?**	3. **What to do in L.A?**

length of question can be a competitive baseline method.

In the experiments, for each topic (city name), we used both our method and the baseline method to rank all the questions from 'SRC-DAT' regarding the topic. Then, we made use of 'SET-B' to evaluate the results.

From Table 4, we see that our method based on either unigram language model or trigram language model outperforms the baseline method significantly. Furthermore, the trigram language model performs better than unigram language model. This might be because the former can take into consideration rich and meaningful word groups, e.g., 'best hotel' or 'place to stay'. These results indicate that our language model-based method has ability to measure utility of questions.

Also, the length normalization has proven to be very effective to meliorate the bias that our language modeling based method prefers to short texts (questions). For both the unigram model and the trigram model, the length normalization boosts the performance by about 20%.

Table 5 shows top 3 results from three different methods: the baseline method, the trigram method and the trigram method with the normalization. In the table, relevant questions are highlighted in bold face.

Using Question Utility as Static Ranking of Question Search. In the experiment, we evaluate the performance of question search with four different configurations: (a) the query likelihood model without static ranking (We used it as our baseline method and refer to it as QM); (b) QM incorporating (question) *utility* scores estimated by the language modeling method (denoted as QM+ULM); (c) QM incorporating (question) *utility* scores estimated by the LexRank method (denoted as QM+LEX); (d) QM incorporating (question) *utility* scores estimated by the combination of the language model method and the LexRank method (denoted as QM+BOTH). For the QM+ULM and the QM+BOTH, we used the trigram language model with length normalization.

Table 6: Comparison of Retrieval Performances

	MAP (△%)	R-prec (△%)	p@5 (△%)
QM	0.489 (-)	0.428 (-)	0.470 (-)
QM+LEX	0.494* (+1.02)	0.443*(3.50)	0.482 (2.55)
QM+ULM	0.509* (4.09)	0.462* (7.94)	0.490 (4.25)
QM+BOTH	0.512* (4.70)	0.469* (9.58)	0.512 (8.93)

Table 6 provides the experimental results. When compared with the baseline method, our methods incorporating question utility consistently show better performance in terms of all the evaluation measures. It can support our hypothesis that question utility can be important factor in question search. All the performance improvements obtained by the use of question utility are statistically significant (t-test, p-value < 0.05).

As shown in Table 6, although both QM+LEX and QM+ULM can improve the performance of question search compared to QM (not using question utility), QM+ULM outperformed QM+LEX (the LexRank method). To our observation, one reason for this can be related to the ability of the language model to capture meaningful word sequences. N-gram language model can naturally reflect important word sequences carrying on askers' intention, for example *'where to stay'* or *'how far from'*. While the LexRank method cannot model such word sequences because it assumes that words in questions are statistically independent from each other. However, the LexRank method is able to reflect similarity between questions into the estimation of question utility, which cannot be achieved by the language model method. Thus, in our experiments, we use the combination (QM+BOTH) of both methods for further improvement of the question utility estimation.

To our observation, our method is especially effective when a query is short and ambiguous. For example, 'best of boston', 'italy travel package', and 'boston to London'. This is because these queries usually have many related results, which is very similar to what happens with web search. In a contrast, our method fails to improve (or even drop) the performance when a query is very specific. For example, 'miles from Los Angeles to New York'.

Table 7 provides the results that are rendered by the baseline method (QM) and our method (QM+ULM). In the table, relevant questions are highlighted in bold face.

Conclusion

In this paper, we studied usefulness of questions within the setting of question search. The contribution of this paper can be summarized in four-fold: (a) we proposed the notion of *question utility* to study usefulness of questions. To the best of our knowledge, this is the first trial of studying usefulness of questions. (b) We proposed to use question utility as static ranking to boost the performance of question search. (c) We

Table 7: Top 5 Results Retrieved by QM and QM + ULM

Query	QM top 5 results	QM + ULM top 5 results
Italy travel package	(1) **Good/Budget Italy Vacation Packages?** (2) Travel packages? (3) Travel package? (4) **Want to know about holiday packages in italy?** (5) Flight travel packages?	(1) **Good/Budget Italy Vacation Packages?** (2) **Want to know about holiday packages in italy?** (3) Travel package? (4) Travel packages? (5) Traveling in Italy?
Boston to New York	New York or Boston? **New York to Boston?** **Boston & new york trip?** **Traveling from Boston to New York?** New York and Boston weather question?	**New York to Boston?** New York or Boston? **Boston & new york trip?** **Traveling from Boston to New York?** **How far is new york from boston?**

examined the language modeling approach, the LexRank algorithm, and their combination in assessing question utility. (d) We conducted a series of experiments on question utility and confirmed the usefulness of question utility as static ranking of question search.

However, our work has a critic. Our experiments have conducted with one restricted domain. For the deep analysis on the effect of utility of a question and our proposed methods, additional experiments on larger and heterogeneous collections are essentially required. It will be our future work.

Acknowledgment

This work was done when the first author was an intern at Microsoft Research Asia.

References

Brin, S., and Page, L. 1998. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems* 30(1–7):107–117.

Burke, R. D.; Hammond, K. J.; Kulyukin, V. A.; Lytinen, S. L.; Tomuro, N.; and Schoenberg, S. 1997. Question answering from frequently asked question files: Experiences with the FAQ finder system. Technical Report TR-97-05.

Erkan, G., and Radev, D. R. 2004. Lexrank: Graph-based centrality as salience in text summarization. *Journal of Artificial Intelligence Research* 22.

Jeon, J.; Croft, W. B.; Lee, J. H.; and Park, S. 2006. A framework to predict the quality of answers with non-textual features. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 228–235.

Jeon, J.; Croft, W. B.; and Lee, J. H. 2005. Finding similar questions in large question and answer archives. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, 84–90.

Jones, K. S.; Walker, S.; and Robertson, S. E. 2000. A probabilistic model of information retrieval: development and comparative experiments - part 2. *Information Processing and Management* 36(6):809–840.

Katz, S. 1997. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics Speech and Signal Processing* 35(3):400–401.

Kleinberg, J. M. 1998. Authoritative sources in a hyperlinked environment. In *SODA '98: Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms*, 668–677. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.

Kurland, O., and Lee, L. 2005. Pagerank without hyperlinks: structural re-ranking using links induced by language models. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 306–313.

Manning, C. D., and Schütze, H. 1999. *Foundations of Statistical Natural Language Processing*. Cambridge, Massachusetts: The MIT Press.

Otterbacher, J.; Erkan, G.; and Radev, D. R. 2005. Using random walks for question-focused sentence retrieval. In *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, 915–922.

Richardson, M.; Prakash, A.; and Brill, E. 2006. Beyond pagerank: machine learning for static ranking. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, 707–715.

Riezler, S.; Vasserman, A.; Tsochantaridis, I.; Mittal, V.; and Liu, Y. 2007. Statistical machine translation for query expansion in answer retrieval. In *ACL '07: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*.

Sneiders, E. 2002. Automated question answering using question templates that cover the conceptual model of the database. In *NLDB '02: Proceedings of the 6th International Conference on Applications of Natural Language to Information Systems-Revised Papers*, 235–239.

Zhai, C., and Lafferty, J. 2001. A study of smoothing methods for language models applied to ad hoc information retrieval. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and Development in Information Retrieval*, 334–342.

Zhou, Y., and Croft, W. B. 2005. Document quality models for web ad hoc retrieval. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, 331–332.