2005

Finding Similar Questions in Large Question and Answer Archives

Jiwoon Jeon
University of Massachusetts Amherst

Finding Similar Questions in Large Question and Answer Archives

Jiwoon Jeon, W. Bruce Croft and Joon Ho Lee
Center for Intelligent Information Retrieval, Computer Science Department
University of Massachusetts, Amherst, MA 01003
[jeon,croft,joonho]@cs.umass.edu

ABSTRACT

There has recently been a significant increase in the number of community-based question and answer services on the Web where people answer other peoples' questions. These services rapidly build up large archives of questions and answers, and these archives are a valuable linguistic resource. One of the major tasks in a question and answer service is to find questions in the archive that a semantically similar to a user's question. This enables high quality answers from the archive to be retrieved and removes the time lag associated with a community-based system. In this paper, we discuss methods for question retrieval that are based on using the similarity between answers in the archive to estimate probabilities for a translation-based retrieval model. We show that with this model it is possible to find semantically similar questions with relatively little word overlap.

Categories and Subject Descriptors

H.3.0 [**Information Search and Retrieval**]: General

General Terms

Algorithms, Measurement, Experimentation

Keywords

Information Retrieval, FAQ retrieval, Language Models

1. INTRODUCTION

One of the emerging trends in Web information service is the growth in sites where people answer other people's questions. This started as digital references services such as the MadSci Network[1] or Ask Dr. Math[2], but has now become a popular part of Web search services on sites such as Google Answers[3] or Wondir[4]. The huge number of retail and business sites that provide a FAQ service can be viewed as the same type of system. Answering questions using a network or community of people is also a very popular Web application in some countries. For example, the question and answer service on Naver, a popular Korean search site, has more than 25,000 questions submitted per day.

While current web search engines enjoy huge commercial success and demonstrate good performance, especially for homepage finding queries, their ability to find relevant information for hard queries such as those asking for opinions or summaries is far from satisfactory. These complicated user information needs can be satisfied by using question and answer (Q&A) services. Another benefit of these services is that users can directly obtain answers rather than a list of potentially relevant documents.

Over time, Q&A services build up very large archives of previous questions and their answers. In order to avoid the lag time involved with waiting for a personal response, a Q&A service will typically automatically search this archive to see if the same question has previously been asked. If the question is found, then a previous answer can be provided with very little delay. In contrast to the usual search paradigm, where the question is used to search the database of potential answers, in this case the question is used to search the database of previous questions, which in turn are associated with answers.

However, measuring semantic similarities between questions is not trivial. Sometimes, two questions that have the same meaning use very different wording. For example, "Is downloading movies illegal?" and "Can I share a copy of a DVD online" have almost the identical meaning but they are lexically very different. Similarity measures developed for document retrieval work poorly when there is little word overlap. This is the same for the traditional and naive sentence distance measures such as the Jaccard coefficient and the overlap coefficient [13].

Three different types of approaches have been developed in the literature to solve this word mismatch problem among questions. The first approach [5] uses knowledge databases such as machine readable dictionaries. However, the quality and structure of current knowledge databases are, based on the results of previous experiments, not good enough for reliable performance. The second approach [19] employs manual rules or templates. These methods are expensive and hard to scale for large size collections. The third approach

[1]http://madsci.org
[2]http://mathforun.org/dr.math/
[3]http://answers.google.com/answers/
[4]http://www.wondir.com

CIKM'05, October 31–November 5, 2005, Bremen, Germany.

[2] is to use statistical techniques developed in information retrieval and natural language processing.

We believe the last approach is the most promising if we have enough training data. A large number of semantically similar but lexically different sentence or question pairs would be an excellent corpus for training, but unfortunately, there has been to date no such collection available on a large scale. Therefore, researchers [15] have used alternative collections that are artificially generated by an approach such as translation of text to a foreign language and then back to the original language. In this paper, we propose an automatic way of building collections of semantically similar question pairs from existing Q&A collections.

After building a collection of similar question pairs, we consider the collection a bilingual corpus and run the IBM machine translation model 1 [4] to learn word translation probabilities. In this case, the word translation probabilities actually denote semantic similarities between words. Given a new question, a translation based information retrieval model exploits the word relationships to retrieve similar questions from Q&A archives. Experimental results show our approach significantly outperforms other baseline retrieval models.

The remainder of the paper is structured as follows. In section 2, we briefly survey related work. Section 3 describes the data collections that we use for our experiments. Section 4 addresses in detail how we find semantically related question pairs. In section 5, we briefly explain IBM model 1 and show some examples of semantic word relationships found using our technique. Section 6 describes the translation model and tests the performance of the model for the task of question retrieval. Section 7 is the conclusion of the paper.

2. RELATED WORK

There has been some research on retrieval using FAQ data. FAQ Finder [5] heuristically combines statistical similarities and semantic similarities between questions to rank FAQs. Conventional vector space models are used to calculated the statistical similarity and WordNet [7] is used to estimate the semantic similarity. Sneiders [19] proposed template based FAQ retrieval systems. These previous approaches were tested with relatively small sized collections and are hard to scale because they are based on specific knowledge databases or handcrafted rules. Lai et al. [10] proposed an approach to automatically mine FAQs from the Web. However, they did not study the use of these FAQs after they were collected.

The archives from Q&A services are different from FAQ collections. Usually, FAQs are created and maintained by experts and the quality of the questions and the answers is good. The number of FAQs in one topic or category are in general less than a few hundred. Q&A archives can be very large and there is no guarantee about the quality of the content. As far as we know, there has been little research done to exploit or search Q&A archives.

Extensive research has been done in the field of question answering [22, 16, 14], but this work is different to the Q&A retrieval task we address in this paper. In question answering, short answers for a relatively limited class of question types are automatically extracted from document collections. In the Q&A retrieval task we address, answers for an unlimited range of questions are retrieved by focusing on finding semantically similar questions in the archive.

Question Title	How to make multi-booting systems?
Question Body	I am using Windows98. I'd like to multi-boot with Windows XP. How can I do this?
Answer	You must partition your hard disk, then install windows98 first. If there is no problem with windows98, then, install windows XP on ... ...

Table 1: A typical question and answer pair in the Naver Q&A archive. The question part is divided into two fields: the question title and the question body. (Translated from Korean)

The idea of finding similar queries using user click logs or retrieval results has been proposed previously [23, 20, 1]. This work assumed that if two different queries have similar click logs or similar retrieval results, then the queries are semantically similar, and the query similarities obtained using this approach would be superior to comparing the text of the queries directly. The results of this work demonstrated the validity of this assumption. We make the similar assumption that if two answers are similar enough then the corresponding questions should be semantically similar.

In this paper, we focus on the lexical chasm problem between questions. Various query expansion techniques have been studied to solve word mismatch problems between queries and documents, including relevance feedback [18], thesaurus-based expansion (e.g. [21]), dimensionality reduction (e.g. [6], [8]), and techniques based on modifying the query based on the top retrieved documents (e.g. [24], [12]). The model proposed here implicitly expands queries using translation probabilities. We generate these translation probabilities for words based on similar question pairs. These translation probabilities are then used in a retrieval model to rank the questions from the archive for a new user-generated question. Berger and Lafferty [3] proposed a formal information retrieval model that integrate word translation probabilities with the language modeling approach [17]. They viewed information retrieval as a statistical translation process. Lavrenko et al. [11] proposed a cross-lingual information retrieval model based on relevance models [12]. This model also can be used with word translation probabilities, but in this paper we focus on the Berger and Lafferty model. We plan to do experiments with the Lavrenko model in future work.

3. COLLECTIONS

Naver[5] is one of the leading portal sites in South Korea and their question and answer service is very popular. Over time, the service has built up a very large archive of questions and answers written in Korean. The experiments in this paper are based on subsets of this archive.

3.1 Question and Answer Archives

Table 1 shows an example question and answer pair in the Naver archive. The question part has two fields - question

[5]http://www.naver.com

title and question body. The question body is an optional field that describes the question title in more detail.

The question title field contains the type of questions that we would expect to receive from users and this data is the basis for the question retrieval experiments. When we refer to the "question" in this paper, in many cases, it is actually referring to the question title. If one question has multiple answers, we merge all the answers into one. The average length of the question title field is 5.8 words, the question body is 49 words, and the answer is 179 words.

We made two different test collections from the archive: the large collection A and the small collection B. Collection A consists of 6.8 million question and answer pairs across all the categories. Collection B has 68,000 question and answer pairs collected from the 'Computer Novice' category. Because of the computational cost, we mainly use collection B for our experiments. Collection A is used to analyze basic properties of the archives and to find the best parameter values for the baseline retrieval models.

3.2 Topics and Relevance Judgements

To verity the performance of the proposed retrieval technique, we need to have sets of topics with relevance judgement information. Two sets of 50 question and answer pairs were randomly selected from the held-out portion of the archives. The first set is for collection A and the pairs are chosen across all the categories. The second set is for collection B, and the pairs are chosen from the 'Computer Novice' category.

Each pair is automatically converted into a topic. The question title becomes a short query, the question body turns into a long query and the answer is converted into a supplemental query. The long queries and the supplemental queries are used only in the relevance judgement procedure. The short queries are used for all other experiments in this paper. The question body has also a role of a description or a narrative of the topic. When the question title is vague, we refer the question body to clarify the meaning of the question.

To find relevant question and answer pairs given a topic, we employ the pooling technique that is used in the TREC[6] conference series. Eighteen different retrieval results were generated by varying the retrieval algorithms, the query type and the search field. Popular retrieval models such as the query-likelihood language model, the Okapi BM25 model and the overlap coefficient are used.

We pooled the top 20 Q&A pairs from each retrieval result and did manual relevance judgments. The correctness of the answer was ignored. As long as the question is semantically identical or very similar to the query, we judge the Q&A pair as relevant. Sometimes, we could not find any relevant Q&A pairs given a topic. In these cases, we manually browsed the collection to find at least one relevant Q&A pair. The final result is a total of 785 relevant Q&A pairs for collection A and 1,557 relevant Q&A pairs for collection B.

3.3 Importance of each field

Previous research [5] implied that similarities between questions are much more important than the similarity between questions and answers in FAQ retrieval tasks. To verify whether this assertion is true with our collections, we carried out some experiments.

[6]http://trec.nist.gov/

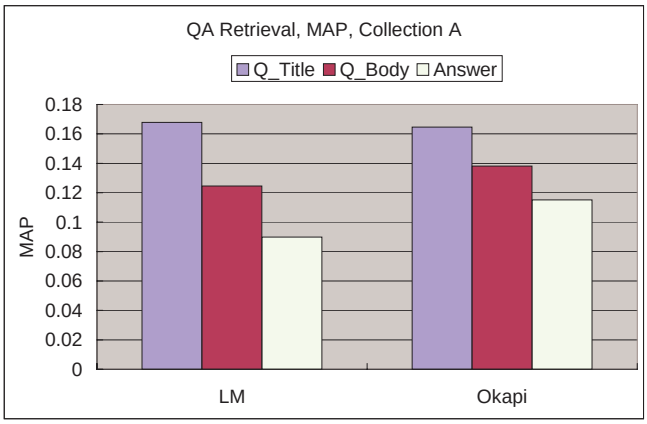

Figure 1: The importance of each field in Q&A retrieval. MAP denotes Mean Average Precision. In both models, the best performance can be achieved by searching the question title field. The answer field is least useful.

Collection A is used in these experiments with the topics described in section 3.2. In the first experiment, we search only the question title field. The second experiment uses only the question body field. The last experiment searches only the answer field. For each experiment, the query-likelihood language model with Dirichlet smoothing and the Okapi BM25 model are used. For each retrieval model, the best parameter values are chosen by exhaustive search of the parameter space.

As Figure 1 shows, regardless of the retrieval model, we can get the best performance by searching the question title field. The performance gaps compared to other fields are significant. The answer field is the least useful.

Even better performance may be achieved by combining all the fields, however, in this paper, we focus only on similarity measures between queries (questions) and question titles. Developing efficient field combination methods will be an issue for future work.

4. GENERATION OF TRAINING SAMPLES

In this section, we describe how we can automatically collect semantically similar question pairs from existing question and answer archives by comparing answers. These pairs serve as training data for our translation based retrieval model.

Many people do not carefully check whether their question has been asked before and post their questions on Q&A boards. Therefore, many semantically identical questions can be found in question and answer archives. Finding an exact duplicate of a question is obviously trivial, but in many cases the same question will have previously been asked but using different wording. Even small lexical differences can make it impossible to retrieve a semantically similar question because questions are short.

To solve this problem, we use similarities between answers to group questions. Our assumption is if two answers are very similar than the corresponding questions should be semantically similar, even though the two questions are lexically very different.

I'd like to insert music into PowerPoint. How can I link sounds in PowerPoint?
How can I shut down my system in Dos-mode. How to turn off computers in Dos-mode.
Photo transfer from cell phones to computers. How to move photos taken by cell phones.
Which application can run bin files? I download a game. How can I execute bin files?
IP tracking method Can I detect the place of a person in anonymous boards using IP addresses?

Table 2: Example question pairs found using the LM-HRANK measure in the collection B. These semantically similar question pairs have little word overlap. (Translated from Korean)

4.1 Algorithm

Initially, we considered four popular document similarity measures to calculate distances between answers: the cosine similarity with the vector space models, the negative KL divergence between the language models, the output score of the query likelihood model and the score of the Okapi model. We found that each measure has its own weakness [9].

The cosine similarity favors short documents and this property become a serious problem in measuring similarities between answers because the lengths of answers vary considerably. Some answers can be very short especially for factoid questions. Other answers are very long because sometimes people generate answers just by copying multiple related documents from the web. Therefore any similarity measure seriously affected by length is not appropriate.

The negative KL divergence in the language modeling framework has shown good performance in document retrieval tasks. However, the values are not symmetric and are not probabilities, so a pair of answers that has a higher negative KL divergence than the other pair does not necessarily have stronger semantic connections. This property makes it hard to rank the pairs. The score of the Okapi model has similar problems.

The score from the query likelihood model is a probability and can be used across different answer pairs but the scores are not symmetric.

Because of the above problems, we found that using ranks instead of scores was more effective. If answer A retrieves answer B at rank r_1 and answer B retrieves answer A at rank r_2, then the similarity between the two answers is defined as the reverse of the harmonic mean of r_1 and r_2. $sim(A,B) = \frac{1}{2}(\frac{1}{r_1} + \frac{1}{r_2})$. We use the query-likelihood language model to calculate the initial ranks. We call this measure LM-HRANK.

4.2 Experiments and Results

A total of $68,000 * 67,999/2$ pairs of answers are possible from 68,000 Q&A pairs in the collection B. All of these pairs are ranked according to the LM-HRANK measure. We empirically set a threshold value (0.005) to judge whether an answer pair is semantically related or not. With a higher threshold value, we will get smaller but better quality collections. To acquire enough training samples, the threshold cannot be too high.

There are 331,965 question pairs that have scores above the threshold. Table 2 shows some of the question pairs found using this method. Each question pair in the examples contains semantically similar questions but shares few common terms.

5. WORD TRANSLATION PROBABILITIES

Having created a collection of similar question pairs, we now need to use this data to estimate word translation probabilities for the proposed retrieval model. We consider the question pair collection a parallel corpus and adopt techniques developed in machine translation to measure the semantic similarities between words.

5.1 Algorithm

The IBM model 1 [4] does not require any linguistic knowledge for the source or the target language and treats every possible word alignment equally. Because of its simplicity and proven performance, we use this model.

In our experiments, the source and the target language is the same. Therefore, the word translation probabilities calculated using the model are actually semantic similarities of words. Any question in a question pair can be a source or a target, so we make two input sentence pairs from a pair of questions by switching the source part and the target part.

In IBM model 1, the translation probability from a source word s to a target word t is given by :

$$P(t|s) = \lambda_s^{-1} \sum_{i=1}^{N} c(t|s; J^i) \quad (1)$$

where λ_s is a normalization factor to make the sum of the translation probabilities add to 1. N is the number of the training samples. In our case, a question pair found in section 4 become a training sample. J^i is the ith pair in the training data.

$$c(t|s; J^i) = \frac{P(t|s)}{P(t|s_1) + \ldots + P(t|s_n)} \#(t, J^i) \#(s, J^i) \quad (2)$$

where $\{s_1, \ldots, s_n\}$ are words in the source sentence in J^i and $\#(t, J^i)$ is the number of times that t occurs in J^i.

As can be seen from the equations, we need the old translation probabilities to estimate the new translation probabilities. We initialize the translation probabilities with random values and then estimate new translation probabilities. This procedure is repeated until the probabilities converge. Brown et al. [4] showed that the procedure always converges to the same final solution regardless of the initial values.

5.2 Experiments and Results

We used the GIZA++[7] toolkit to learn the IBM model. After removing stop words, the collection of the 331,965 question pairs duplicated by switching the source part and the target part and then used as input for the toolkit. Table 3 shows the top 10 words that are most similar to the given words.

In many cases, the most similar word to a given word is the word itself. Most of the words in the table are semantically related words to the given words. The algorithm discovers various semantic relationships between words. For example, in the first column, we can see the graphic file format 'bmp'

[7]http://www.fjoch.com/GIZA++.html

Rank	bmp	format	music	intel	excel	font	watch	memory
1	bmp	format	music	pentium	excel	font	watch	memory
2	jpg	format*	file	4	korean	korean	time	virtual
3	gif	xp	tag	celeron	function	97	background	shortage
4	save	windows	sound	amd	novice	add	start	ram
5	file	hard	background	intel	cell	download	date	message
6	picture	98	song	performance	disappear	control-panel	display	configuration
7	change	partition	play	support	convert	register	tray	256
8	ms-paint	drive	mp3	question	if	install	power	extend
9	convert	disk	cd	buy	xls	default	screen	system
10	photo	C	source	cpu	record	photoshop	wrong	windows

Table 3: The first low shows the source words and each column shows top 10 words that are most semantically similar to the source word. It is not hard to notice most of the words in the table have somewhat strong semantic relationship with the source words. (format and format* are different in Korean but both words are translated into 'format' in English) (Translated from Korean)

is closely related to other common graphic file formats such as 'jpg' and 'gif'. We can also notice 'bmp' is semantically connected to verbs such as 'save' and 'convert'.

6. QUESTION RETRIEVAL

In this section, we show how we can bridge the lexical chasm between question titles using the word translation probabilities that we calculate in previous section. Berger and Lafferty viewed information retrieval as statistical translation and proposed a translation based information retrieval model that exploits word translation probabilities in the language modeling framework. Obviously, the success of the model depends on the quality of the word translation probabilities.

6.1 Translation Model

In the language modelling framework, the similarity between a query and a document is given by the probability of the generating the query from the document language model. Usually, i.i.d sampling and unigram document language models are used.

$$sim(Q, D) \approx P(Q|D) = \prod_{w \in Q} P(w|D) \quad (3)$$

To avoid zero probabilities and estimate more accurate language models, documents are smoothed using a background collection,

$$P(w|D) = (1 - \lambda)P_{ml}(w|D) + \lambda P_{ml}(w|C) \quad (4)$$

$P_{ml}(w|C)$ is the probability that the term w is generated from the collection C. $P_{ml}(w|C)$ is estimated using the maximum likelihood estimator. λ is the smoothing parameter. In the translation model, $P_{ml}(w|D)$ in equation (4) is replaced by $\Sigma_{t \in D} T(w|t)P_{ml}(t|D)$.

$$P(w|D) = (1 - \lambda) \sum_{t \in D} (T(w|t)P_{ml}(t|D)) + \lambda P_{ml}(w|C) \quad (5)$$

$T(w|t)$ denotes the probability that word w is the translation of word t.

In our experiments, we assume the probability of self-translation is always 1, $T(w|w) = 1$. There is no theoretical justification for this assumption but this modification empirically gives better retrieval performances in our experiments. If we use the original model, sometimes, the importance of matching terms are lowered because of the low self-translation probabilities.

6.2 Experiments and Results

We used the translation model to retrieve relevant questions given 50 short queries in the topics produced for the collection B. We search only the question title fields. Similarities between the query question and the question titles in the collection B are calculated.

We compare the performance of the translation model with three different baseline retrieval models; vector space model with the cosine similarity, the Okapi BM25 model and the query-likelihood language model.

For each baseline experiments, we use the parameter values that are optimal in the collection A. The only parameter in the translation model is the smoothing parameter, and for this we use the same parameter value that is used for the baseline query-likelihood language model.

Table 4 and Figure 2 show the evaluation results of the experiments. Figure 2 show 11pt recall and precision graphs. As can be seen from the graphs, our approach outperforms other baseline models at all recall levels. While the vector space model with the cosine similarity works poorly, the query likelihood language model and the Okapi model show comparable performance to each other. Table 4 shows various evaluation measures such as MAP (Mean Average Precision) and R-precision. In all evaluation measures, our approach outperforms the other models. We did statistical significance tests using the two-tailed sign test at confidence level 95%. The performance improvements are statistically significant with all evaluation measures.

6.3 Examples and Analysis

Table 5 explains some of the reasons why the translation model works better than other models in our experiments. In example (a), the query and the question title have almost the same meaning and the translation model retrieves the question at rank 10. Other models fail to retrieve the question even in the top 1000 ranks because they could not capture the semantic relationship between 'burned' and 'recorded' or 'read' and 'recognize'.

In the second example (b), because of some segmentation errors in the morphological processing routines 'cpu100%' becomes an index term and the baseline models could not recognize 'cpu100%' and 'cpu 100%' as the same thing. For-

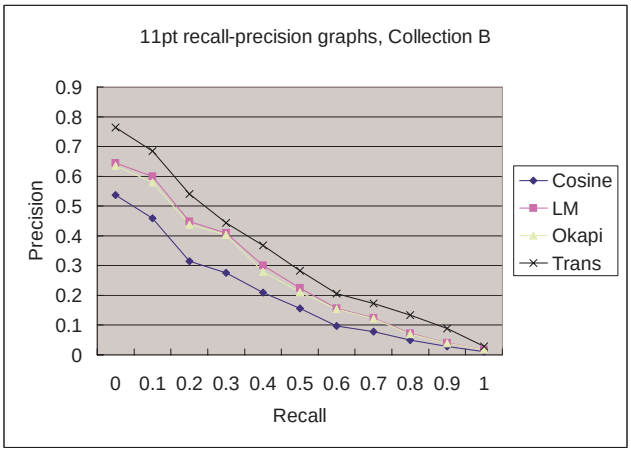

Figure 2: Comparison of the question retrieval performance, 11 points recall-precision graphs. The translation model outperforms other models in all recall area. The query-likelihood language model(LM) and the BM25 Okapi model(Okapi) show similar performances and the cosine similarity is the worst.

Model	Cosine	LM	Okapi	Trans
MAP	0.183	0.258	0.251	**0.314**
R-Precision at 5	0.368	0.492	0.476	**0.520**
R-Precision at 10	0.310	0.456	0.436	**0.480**

Table 4: Comparison of the question retrieval performance. In all measures the translation model works best. The performance gap is statistically significant(Two-tailed sign test with confidence level 95%). The cosine similarity works poorly and the query-likelihood language model(LM) and the BM25 Okapi model(Okapi) show comparable results.

tunately, there are a few training question pairs that contain 'cpu100%' in one side and 'cpu 100%' in the other side. From these pairs, the IBM model learns that 'cpu' and 'cpu100%' is related, with a translation probability $T(cpu100\%|cpu) = 0.237$. The translation model exploits this word relationship and successfully retrieves the question at rank 10. Example (c) is a similar case where the baseline models fail to catch the relationship between 'JuHyunTech' and 'JuHyun'.

We also find that sometimes our approach can successfully retrieve relevant questions even if the questions contain misspelled query terms. For example, some questions containing 'ourlook' by mistake can be retrieved by a query having a word 'outlook'. These examples show our approach can address a variety of lexical disagreement problems.

7. CONCLUSION AND FUTURE WORK

In this paper, we show that a question and answer archive from a community-based Q&A service can serve as a valuable resource to train retrieval models that can recognize semantically similar questions. Specifically, we showed that a retrieval model based on translation probabilities learned from the archive significantly outperforms other approaches in terms of finding semantically similar questions despite a considerable amount of lexical mismatch.

Because of the computational cost, we initially used a relatively small subset of the available archive. As we increase the number of the training samples, we expect to get more accurate word translation probabilities and better retrieval performance. Instead of the IBM model 1, we also plan to study more advanced techniques that exploit more knowledge such as part of speech tagging information and word alignment that may increase the accuracy of our system. We also plan to use the translation probabilities learned from the Q&A archive for document retrieval experiments.

8. ACKNOWLEDGEMENTS

This work was supported by NHN Corp., the Center for Intelligent Information Retrieval and NSF grant number DUE-0226144. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s)' and do not necessarily reflect those of the sponsor.

9. REFERENCES

[1] D. Beeferman and A. Berger. Agglomerative clustering of a search engine query log. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 407–416, 2000.

[2] A. Berger, R. Caruana, D. Cohn, D. Freitag, and V. Mittal. Bridging the lexical chasm: statistical approaches to answer-finding. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 192–199, 2000.

[3] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 222–229, 1999.

[4] P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer. The mathematics of statistical machine translation: parameter estimation. *Comput. Linguist.*, 19(2):263–311, 1993.

[5] R. D. Burke, K. J. Hammond, V. A. Kulyukin, S. L. Lytinen, N. Tomuro, and S. Schoenberg. Question answering from frequently asked question files: Experiences with the faq finder system. Technical report, 1997.

[6] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[7] C. Fellbaum. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

[8] T. Hofmann. Probabilistic latent semantic analysis. In *Proceedings of Uncertainty in Artificial Intelligence, UAI'99*, pages 289–296, 1999.

[9] J. Jeon, W. B. Croft, and J. H. Lee. Finding semantically similar questions based on their answers. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 617–618, 2005.

[10] Y.-S. Lai, K.-A. Fung, and C.-H. Wu. Faq mining via list detection. In *Proceedings of the Workshop on*

(a)

Query	Can't read a burned CD	LM Rank	Trans Rank
Question	I recorded a movie on a CD but it is not recognized	> 1000	10
Analysis	T(burned|recorded) = 0.076, T(read|recognized) = 0.1		

(b)

Query	A question about cpu100%	LM Rank	Trans Rank
Question	Problem of cpu 100%	> 1000	10
Analysis	T(cpu100%|cpu) = 0.237, T(cpu100%|100%) = 0.177		

(c)

Query	Please review JuHyunTech computers' specifications.	LM Rank	Trans Rank
Question	Review configurations of JuHyun Tech computers	368	5
Analysis	T(JuHyunTech|JuHyun) = 0.078		

Table 5: Analysis of the question retrieval examples. The 'LM Rank' is the rank of the question in the retrieval results of the query-likelihood language model. '> 1000' denotes the question is ranked outside of the rank 1000. The 'Trans Rank' is the rank of the question in the translation model. The 'Analysis' fields show some important word translation probabilities that affect the retrieval results. (Translated from Korean)

Multilingual Summarization and Question Answering, 2002.

[11] V. Lavrenko, M. Choquette, and W. B. Croft. Cross-lingual relevance models. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 175–182, 2002.

[12] V. Lavrenko and W. B. Croft. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127, 2001.

[13] C. Manning and H. Schutze. *Foundation of statistical natural language processing*. The MIT Press, 1999.

[14] D. Metzler and W. B. Croft. Analysis of statistical question classification for fact-based questions. *Information Retrieval*, 8(3):481–504, 2005.

[15] V. Murdock and W. B. Croft. Simple translation models for passage retrieval in factoid question answering. In *Proceedings of the Workshop on Information Retrieval for Question Answering*, 2004.

[16] M. A. Pasca and S. M. Harabagiu. High performance question/answering. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 366–374, 2001.

[17] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281, 1998.

[18] J. J. Rocchio. Relevance feedback in information retrieval. In *The SMART Retrieval System: Experiments in Automatic Indexing*, pages 324–336. Prentice Hall, 1971.

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

[20] A. Tombros, R. Villa, and C. J. V. Rijsbergen. The effectiveness of query-specific hierarchic clustering in information retrieval. *Inf. Process. Manage.*, 38(4):559–582, 2002.

[21] E. M. Voorhees. Query expansion using lexical-semantic relations. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 61–69, 1994.

[22] E. M. Voorhees. Overview of the TREC 2004 question answering track. In *Proceedings of the Thirteenth Text Retrieval Conference*, 2004.

[23] J. R. Wen, J. Y. Nie, and H. Zhang. Query clustering using user logs. *ACM Trans. Inf. Syst.*, 20(1):59–81, 2002.

[24] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4–11, 1996.