Statistical Machine Translation Models for Personalized Search

Rohini U *
AOL India R& D
Bangalore, India
Rohini.uppuluri@corp.aol.com

Vamshi Ambati
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, USA
vamshi@cs.cmu.edu

Vasudeva Varma
LTRC, IIIT Hyd
Hyderabad, India
vv@iiit.ac.in

Abstract

Web search personalization has been well studied in the recent few years. Relevance feedback has been used in various ways to improve relevance of search results. In this paper, we propose a novel usage of relevance feedback to effectively model the process of query formulation and better characterize how a user relates his query to the document that he intends to retrieve using a noisy channel model. We model a user profile as the probabilities of translation of query to document in this noisy channel using the relevance feedback obtained from the user. The user profile thus learnt is applied in a re-ranking phase to rescore the search results retrieved using an underlying search engine. We evaluate our approach by conducting experiments using relevance feedback data collected from users using a popular search engine. The results have shown improvement over baseline, proving that our approach can be applied to personalization of web search. The experiments have also resulted in some valuable observations that learning these user profiles using snippets surrounding the results for a query gives better performance than learning from entire document collection.

1 Introduction

Most existing text retrieval systems, including the web search engines, suffer from the problem of "one size fits all": the decision of which documents to retrieve is made based only on the query posed, without consideration of a particular user's preferences and search context. When a query (e.g. "jaguar") is ambiguous, the search results are inevitably mixed in content (e.g. containing documents on the jaguar cat and on the jaguar car), which is certainly non-optimal for a given user, who is burdened by having to sift through the mixed results. In order to optimize retrieval accuracy, we clearly need to model the user appropriately and personalize search according to each individual user. The major goal of personalized search is to accurately model a user's information need and store it in the user profile and then re-rank the results to suit to the user's interests using the user profile. However, understanding a user's information need is, unfortunately, a very difficult task partly because it is difficult to model the search process which is a cognitive process and partly because it is difficult to characterize a user and his preferences and goals. Indeed, this has been recognized as a major challenge in information retrieval research (et. al, 2003).

In order to address the problem of personalization one needs to clearly understand the actual process of search. First the user has an information need that he would like to fulfill. He is the only entity in the process that knows the exact information he needs and also has a vague notion of the document that can full fill his specific information need. A query based search engine is at his disposal for identifying this particular document or set of documents from among a vast repository of them. He then formulates a query that he thinks is congruent to the document he imagines to fulfill his need and poses it to the search engine. The search engine now returns

*This work was done when the first and second authors were at IIIT Hyderabad, India.

a list of results that it calculates as relevant according to its ranking algorithm. Every user is different and has a different information need, perhaps overlapping sometimes. The way a user conceives an ideal document that fulfills his need also varies. It is our hypothesis that if one can learn the variations of each user in this direction, effective personalization can be done.

Most approaches to personalization have tried to model the user's interests by requesting explicit feedback from the user during the search process and observing these relevance judgments to model the user's interests. This is called relevance feedback, and personalization techniques using it have been proven to be quite effective for improving retrieval accuracy (Salton and Buckley, 1990; Rocchio, 1971). These approaches to personalization have considered, user profile to be a collection of words, ontology, a matrix etc.

We use relevance feedback for personalization in our approach. However we propose a novel usage of relevance feedback to effectively model the process of query formulation and better characterize how a user relates his query to the document that he intends to retrieve as discussed in the web search process above. A user profile learnt from the relevance feedback that captures the query generation process is used as a guide to understand user's interests over time and personalize his web search results.

Interestingly, a new paradigm has been proposed for retrieval rooted from statistical language modeling recently that views the query generation process through a Noisy channel model (Berger and Lafferty, 1999) . It was assumed that the document and query are from different languages and the query generation process was viewed as a translation from the document language which is more verbose to the language of the query which is more compact and brief. The noisy channel model proposed by Berger and Lafferty (Berger and Lafferty, 1999) inherently captures the dependencies between the query and document words by learning a translation model between them. As we intend to achieve personalized search by personalizing the query formulation process, we also perceive the user profile learning through a Noisy Channel Model. In the model, when a user has an information need, he also has an ideal document in mind that fulfills his need. The user tries to in a way translate the notion of the ideal document into a query that is more compact but congruent to the document. He then poses this query to the search engine and retrieves the results. By observing this above process over time, we can capture how the user is generating a query from his ideal document. By learning this model of a user, we can predict which document best describes his information need for the query he poses. This is the motive of personalization. In our approach, we learn a user model which is probabilistic model for the noisy channel using statistical translation approaches and from the past queries and their corresponding relevant documents provided as feedback by the user.

The rest of the paper is organized as follows. We first describe the related work on personalized search then we provide the background and the framework that our approach is based upon. we discuss the modeling of a user profile as a translation model. after which we describe applying it to personalized search. we describe our experimental results followed by conclusions with directions to some future work.

2 Related Work

There has been a growing literature available with regard to personalization of search results. In this section, we briefly overview some of the available literature.

(Pretschner and Gauch, 1999) used ontology to model users interests, which are studied from users browsed web pages. (Speretta and Gauch, 2004) used users search history to construct user profiles. (Liu et al., 2002) performed personalized web search by mapping a query to a set of categories using a user profile and a general profile learned from the user's search history and a category hierarchy respectively. (Hatano and Yoshikawa., 2004) considered the unseen factors of the relationship between the web users behaviors and information needs and constructs user profiles through a memory-based collaborative filtering approach.

To our knowledge, there has been a very little work has been done that explicitly uses language models to personalization of search results. (Croft et al., 2001) discuss about relevance feedback and

query expansion using language modeling. (Shen et al., 2005) use language modeling for short term personalization by expanding queries.

Earlier approaches to personalization have considered, user profile to be a collection of words, ontology, language model etc. We perceive the user profile learning through a Noisy Channel Model. In the model, when a user has an information need, he also has a vague notion of what is the ideal document that he would like to retrieve. The user then creates a compact query that he thinks would retrieve the document. He then poses the query to the search engine. By observing this above process over time, we learn a user profile as the probabilities of translation for the noisy channel that converts his document to the query. We then use this profile in re-ranking the results of a search engine to provide personalized results.

3 Background

In this section, we describe the statistical language modeling and the translation model framework for information retrieval that form a basis for our research.

The basic approach for language modeling for IR was proposed by Ponte and Croft (Ponte and Croft, 1998). It assumes that the user has a reasonable idea of the terms that are likely to appear in the ideal document that can satisfy his/her information need, and that the query terms the user chooses can distinguish the ideal document from the rest of the collection. The query is thus generated as the piece of text representative of the ideal document. The task of the system is then to estimate, for each of the documents in the collection, which is most likely to be the ideal document.

$$\arg\max_D P(D|Q) = \arg\max_D P(Q|D)P(D)$$

where Q is a query and D is a document. The prior probability $P(D)$ is usually assumed to be uniform and a language model $P(Q|D)$ is estimated for every document. In other words, they estimate a probability distribution over words for each document and calculate the probability that the query is a sample from that distribution. Documents are ranked according to this probability. The basic model has been extended in a variety of ways. Modeling documents as in terms of a noisy channel model by Berger & Lafferty (Berger and Lafferty, 1999), mixture of topics, and phrases are considered (Song and Croft., 1999), (Lavrenko and Croft, 2001) explicitly models relevance, and a risk minimization framework based on Bayesian decision theory has been developed (Zhai and Lafferty, 2001).

The noisy channel by Berger and Lafferty (Berger and Lafferty, 1999) view a query as a distillation or translation from a document describing the query generation process in terms of a noisy channel model. In formulating a query to a retrieval system, a user begins with an information need. This information need is then represented as a fragment of an "ideal document", a portion of the type of document that the user hopes to receive from the system. The user then translates or "distills" this ideal document fragment into a succinct query, selecting key terms and replacing some terms with related terms.

To determine the relevance of a document to a query, their model estimates the probability that the query would have been generated as a translation of that document. Documents are then ranked according to these probabilities. More specifically, the mapping from a document term w to a query term q_i is achieved by estimating translation models $P(q|w)$. Using translation models, the retrieval model becomes

$$P(Q|D) = \prod_{q_i \in Q} \alpha P(q_i|GE) + (1-\alpha) \sum_{w \in D} P(q_i|w)P(w|D)$$

where $P(q_i|GE)$ is the smoothed or general probability obtained from a large general corpus. $P(q_i|w)$ is an entry in the translation model. It represents the probability of generation of the query word q_i for a word w in the document. $P(w|D)$ is the probability of the word w in the document and α is a weighting parameter which lies between 0 and 1.

4 User Profile as a Translation Model

We perceive the user profile learning as learning the channel probabilities of a Noisy Channel Model that generates the query from the document. In the model, when a user has an information need, he also has a vague notion of what is the ideal document that he would like to retrieve. The user then creates

a compact query that he thinks would retrieve the document. He then poses the query to the search engine. By observing this above process over time, we can learn how the user is generating a query from his notion of an ideal document. By learning this, we can predict which document best describes his information need. The learnt model, called a user profile, is thus capable of personalizing results for that particular user. Hence, the user profile here is a translation model learnt from explicit feedback of the user using statistical translation approaches. Explicit feedback consists of the past queries and their corresponding relevant documents provided as feedback by the user. A translation model is a probabilistic model consisting of the triples, the source word, the target word and the probability of translation. The translation model here is between document words and queries words. Therefore the user profile as a translation model in our approach will consist of triples of a document word, a query word and the probability of the document word generating the query word.

5 Personalized Search

In this section, we describe how we perform personalized search using the proposed translation model based user profile. First, a user profile is learnt using the translation model process then the re-ranking is done using the learnt user profile.

5.1 Learning user profile

In our approach, a user profile consists of a statistical translation model. A translation model is a probabilistic model consisting of the triples, the source word, the target word and the probability of translation. Our user profiles consists of the following triples, a document word, a query word and the probability of the document word generating the query word.

Consider a user u, let $\{ \{Q_i, D_i\}, i = 1, 2, ..., N\}$ represent the past history of the user u. where Q_i is the query and D_i is the concatenation of all the relevant documents for the query Q_i and let $D_i = \{w_1, w_2, ..., w_n\}$ be the words in it. The user profile learnt from the past history of user consists of the following triples of the form $(q, w_i, p(q|w_i))$ where q is a word in the query Q_i and w_i is a word in the document D_i.

Translation model is typically learnt from parallel texts i.e a set of translation pairs consisting of source and target language sentences. In learning the user profile, we first extract parallel texts from the past history of the user and then learn the translation model which is essentially the user profile. In the subsections below, we describe the process in detail.

5.1.1 Extracting Parallel Texts

By viewing documents as samples of a verbose language and the queries as samples of a concise language, we can treat each document-query pair as a translation pair, i.e. a pair of texts written in the verbose language and the concise language respectively. The extracted parallel texts consists of pairs of the form $\{Q_i, D_{rel}\}$ where D_{rel} is the concatenation of contexts extracted from all relevant document for the query Q_i.

We believe that short snippets extracted in the context of the query would be better candidates for D_{rel} than using the whole document. This is because there can be a lot of noisy terms which need not right in the context of the query. We believe a short snippet usually N (we considered 15) words to the left and right of the query words, similar to a short snippet displayed by search engines can better capture the context of the query. In deed we experimented with different context sizes for D_{rel}. The first is using the whole document i.e., considering the query and concatenation of all the relevant documents as a pair in the parallel texts extracted which is called $D_{documents}$ The second is using just a short text snippet from the document in the context of query instead of the whole document which is called $D_{snippets}$ Details are described in the experiments section.

5.1.2 Learning Translation Model

According to the standard statistical translation model (Brown et al., 1993), we can find the optimal model M^* by maximizing the probability of generating queries from documents or

$$M^* = \arg\max_M \prod_{i=1}^{N} P(Q_i|D_i, M)$$

qw	dw	P(qw|dw,u)
journal	kdd	0.0176
journal	conference	0.0123
journal	journal	0.0176
journal	sigkdd	0.0088
journal	discovery	0.0211
journal	mining	0.0017
journal	acm	0.0088
music	music	0.0375
music	purchase	0.0090
music	mp3	0.0090
music	listen	0.0180
music	mp3.com	0.0450
music	free	0.0008

Table 1: Sample user profile

To find the optimal word translation probabilities $P(qw|dw, M*)$, we can use the EM algorithm. The details of the algorithm can be found in the literature for statistical translation models, such as (Brown et al., 1993).

IBM Model1 (Brown et al., 1993) is a simplistic model which takes no account of the subtler aspects of language translation including the way word order tends to differ across languages. Similar to earlier work (Berger and Lafferty, 1999), we use IBM Model1 because we believe it is more suited for IR because the subtler aspects of language used for machine translation can be ignored for IR. GIZA++ (Och and Ney, 2003), an open source tool which implements the IBM Models which we have used in our work for computing the translation probabilities. A sample user profile learned is shown in Table 1.

5.2 Re-ranking

Re-ranking is a phase in personalized search where the set of documents matching the query retrieved by a general search engine are re-scored using the user profile and then re-ranked in descending order of rank of the document. We follow a similar approach in our work.

Let $\mathcal{D}$ be set of all the documents returned by the search engine. The rank of each document D returned for a query Q for user u is computing using his user profile as shown in Equation 1.

$$P(Q|D,u) = \prod_{q_i \in Q} \alpha P(q_i|GE) + (1-\alpha) \sum_{w \in D} P(q_i|w,u) P(w|D) \quad (1)$$

where $P(q_i|GE)$ is the smoothed or general probability obtained from a large general corpus. $P(q_i|w,u)$ is an entry in the translation model of the user. It represents the probability of generation of the query word q_i for a word w in the document. $P(w|D)$ is the probability of the word w in the document and α is a weighting parameter which lies between 0 and 1.

6 Experiments

We performed experiments evaluating our approach on data set consisting of 7 users. Each user submitted a number of queries to a search engine (Google). For each query, the user examined the top 10 documents and identified the set of relevant documents. Table 2 gives the statistics of the data sets. There is no repetition of query for any user though repetition of some words in the query exists (see Table 2). The document collection consists of top 20 documents from google which is actually the set of documents seen by the user while accessing the relevance of the documents. In all, the total size of the document collection was 3,469 documents. We did not include documents of type doc and pdf files.

To evaluate our approach, we use the 10-fold cross-validation strategy (Mitchell, 1997). We divide the data of each user into 10 sets each having (approximately) equal number of search queries (For example, for user1 had 37 queries in total, we divided this into 10 sets with 4 queries each approximately). Learning of user profile is done 10 times, each time leaving out one of the sets from training, but using only the omitted subset for testing. Performance is computed in the testing phase for each time and average of the 10 times is taken. In the testing phase, we take each query and re rank the results using the proposed approach using his profile learned from nine other sets. For measuring performance for each query, we compute Precision @10 (P@10), a widely used metric for evaluating personalized search algorithms. It is defined as the proportion of relevant documents among the top 10 results for the given ranking of documents. P@10 is computed by comparing with the relevant documents present in the data. All the values presented in the tables are average values which are averaged over all queries for each user, unless otherwise specified. We used Lucene[1], an open source search engine as the general search engine to first retrieve a

[1] http://lucene.apache.org

User	No. Q	% of Unique words in Q	Total Rel	Avg. Rel
1	37	89	236	6.378
2	50	68.42	178	3.56
3	61	82.63	298	4.885
4	26	86.95	101	3.884
5	33	80.76	134	4.06
6	29	78.08	98	3.379
7	29	88.31	115	3.965

Table 2: Statistics of the data set of 7 users

Set	Contextless	Proposed
User1	0.1433	0.1421
User2	0.1426	0.2445
User3	0.1016	0.1216
User4	0.0557	0.1541
User5	0.1877	0.3933
User6	0.1566	0.3941
User7	0.1	0.1833
Avg	0.1268	**0.2332**

Table 3: Precision @10 results for 7 users

set of results matching the query.

6.0.1 Comparison with Contextless Ranking

We test the effectiveness of our user profile by comparing with a contextless ranking algorithm. We used a generative language modeling for IR as the context less ranking algorithm (Query Likelihood model (Ponte and Croft, 1998; Song and Croft., 1999)). This is actually the simplest version of the model described in Equation 1. Each word w can be translated only as itself that is the translation probabilities (see Equation 1) are "diagonal".

$$P(q_i|w,u) = \begin{cases} 1 & \text{if } q = w \\ 0 & \text{Otherwise} \end{cases}$$

This serves as a good baseline for us to see how well the translation model actually captured the user information. For fair testing similar to our approach, for each query, we first retrieve results matching a query using a general search engine (Lucene). Then we rank the results using the formula shown in Equation 2.

$$P(Q|D) = \prod_{q_i \in Q} \alpha P(q_i|GE) + (1-\alpha) P(q_i|D) \quad (2)$$

We used IBM Model1 for learning the translation model (i.e., the user profile). The general English probabilities are computed from all the documents in the lucene's index. Similar to earlier works (Berger and Lafferty, 1999), we simply set the value of α to be 0.05. The values reported are P@10 values average over all 10 sets and the queries for the respective user. Table 3 clearly shows the improvement brought in by the user profile.

6.0.2 Experiments with Different Models

We performed an experiment to see if different training models for learning the user profile affected the performance. We experimented with two models. The first is a basic model and used in earlier work, IBM Model1. The second is using the GIZA++ default parameters. We observed that user profile learned using IBM Model1 outperformed that using GIZA++ default parameters. We believe this is because, IBM Model1 is more suited for IR because the subtler aspects of language used for machine translation (which are used in GIZA++ default parameters) can be ignored for IR. We obtained an average P@10 value of 0.2333 for IBM Model1 and 0.2075 for GIZA++.

Training Model	Document Test	Snippet Test
IBM Model1		
Document Train	0.2062	0.2028
Snippet Train	0.2333	**0.2488**
GIZA++		
Document Train	0.1799	0.1834
Snippet Train	0.2075	0.2034

Table 4: Summary of Comparison of different Models and Contexts for learning user profile

6.0.3 Snippet Vs Document

In extracting parallel texts consists of pairs of the form $\{Q_i, D_{rel}\}$ where D_{rel} is the concatenation of contexts extracted from all relevant document for the query Q_i we experimented with different context sizes for D_{rel}.

We believe that a short snippet extracted in the context of the query would be better candidate for D_{rel} than using the whole document. This is because there can be a lot of noisy terms which need not useful in the context of the query. We believe a short snippet usually N (we considered 15) words to the left and right of the query words, similar to a short snippet displayed by search engines can better

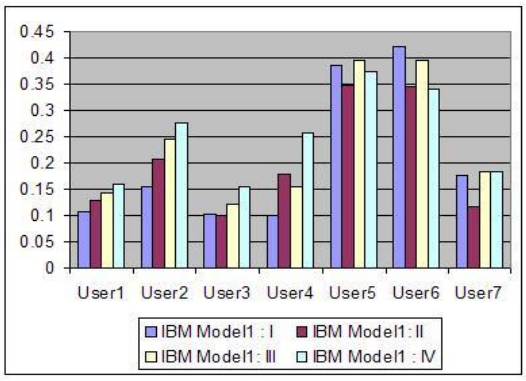

Figure 1: Comparison of Snippet Vs Document Training using IBM Model1 for training.
IBM Model1 : I - Document Training and Document Testing, IBM Model1 : II - Document Training and Snippet Testing, IBM Model1 : III - Snippet Training and Document Testing, IBM Model1 : IV - Snippet Training and Snippet Testing

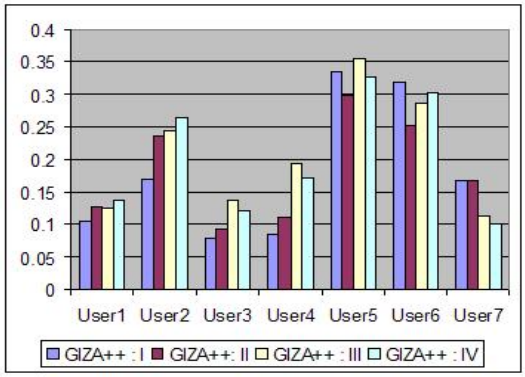

Figure 2: Comparison of Snippet Vs Document Training using GIZA++ Default parameters for training.
GIZA++:I - Document Training and Document Testing, GIZA++:II - Document Training and Snippet Testing, GIZA++:III - Snippet Training and Document Testing, GIZA++:IV - Snippet Training and Snippet Testing

capture the context of the query.

We experimented with two context sizes. The first is using the whole document i.e., considering the query and concatenation of all the relevant documents as a pair in the parallel texts extracted which is called $D_{documents}$. The second is using just a short text snippet from the document in the context of query instead of the whole document which is called $D_{snippets}$. The user profile learning from pairs of parallel texts $\{Q, D_{documents}\}$ is called *Document Train*. The user profile learning from pairs of parallel texts $\{Q, D_{snippets}\}$ is called *Snippet Train*. The user profiles are trained using both IBM Model1 and GIZA++ and comparison of the two is shown in Table 4.

We also experimented with the size of the context used for testing. Using the document for re-ranking as shown in Equation 1 (called *Document Test*) [2] and using just a short snippet extracted from the document for testing (called *Snippet Test*). Table 4 shows the average P@10 over the 10 sets and all queries and users.

We observed that, not only did the model used for training affected P@10, but also the data used in training and testing, whether it was a snippet or document, showed a large variation in the performance. Training using IBM Model1 using the snippet and testing using snippet achieved the best results. This is in agreement with the discussion that the snippet surrounding the query captures the context of the query better than a document which may contain many words that could possibly be unrelated to the query, therefore diluting the strength of the models learnt. The detailed results for all the users are shown in Figure 1 and Figure 2.

[2]It is to be noted that *Snippet Train* and *Document Test* and training using IBM Model1 is the default configuration used for all the reported results unless explicitly specified.

7 Conclusions and Future Work

Relevance feedback from the user has been used in various ways to improve the relevance of the results for the user. In this paper we have proposed a novel usage of relevance feedback to effectively model the process of query formulation and better characterize how a user relates his query to the document that he intends to retrieve. We applied a noisy channel model approach for the query and the documents in a retrieval process. The user profile was modeled using the relevance feedback obtained from the user as the probabilities of translation of query to document in this noisy channel. The user profile thus learnt was applied in a re-ranking phase to rescore the search results retrieved using general information retrieval models. We evaluate the usage of our approach by conducting experiments using relevance feedback data collected from users of a popular search engine. Our experiments have resulted in

some valuable observations that learning these user profiles using snippets surrounding the results for a query show better performance than when learning from entire documents. In this paper, we have only evaluated explicit relevance feedback gathered from a user and performed our experiments. As part of future work, we would like to evaluate our approach on implicit feedback gathered probably as click-through data in a search engine, or on the client side using customized browsers.

References

Adam Berger and John D. Lafferty. 1999. Information retrieval as statistical translation. In *Research and Development in Information Retrieval*, pages 222–229.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Comput. Linguist.*, 19(2):263–311.

W. Bruce Croft, Stephen Cronen-Townsend, and Victor Larvrenko. 2001. Relevance feedback and personalization: A language modeling perspective. In *DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*.

Jamie Allan et. al. 2003. Challenges in information retrieval language modeling. In *SIGIR Forum*, volume 37 Number 1.

K. Sugiyama K. Hatano and M. Yoshikawa. 2004. Adaptive web search based on user profile constructed without any effort from users. In *Proceedings of WWW 2004*, page 675 684.

Victor Lavrenko and W. Bruce Croft. 2001. Relevance-based language models. In *Research and Development in Information Retrieval*, pages 120–127.

F. Liu, C. Yu, and W. Meng. 2002. Personalized web search by mapping user queries to categories. In *Proceedings of the eleventh international conference on Information and knowledge management, ACM Press*, pages 558–565.

Tom Mitchell. 1997. *Machine Learning*. McGrawHill.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Jay M. Ponte and W. Bruce Croft. 1998. A language modeling approach to information retrieval. In *Research and Development in Information Retrieval*, pages 275–281.

A. Pretschner and S. Gauch. 1999. Ontology based personalized search. In *ICTAI.*, pages 391–398.

J. J. Rocchio. 1971. Relevance feedback in information retrieval, the smart retrieval system. *Experiments in Automatic Document Processing*, pages 313–323.

G. Salton and C. Buckley. 1990. Improving retrieval performance by relevance feedback. *Journal of the American Society of Information Science*, 41:288–297.

Xuehua Shen, Bin Tan, and Chengxiang Zhai. 2005. Implicit user modeling for personalized search. In *Proceedings of CIKM 2005*.

F. Song and W. B. Croft. 1999. A general language model for information retrieval. In *Proceedings on the 22nd annual international ACM SIGIR conference*, page 279280.

Micro Speretta and Susan Gauch. 2004. Personalizing search based on user search histories. In *Thirteenth International Conference on Information and Knowledge Management (CIKM 2004)*.

Chengxiang Zhai and John Lafferty. 2001. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of ACM SIGIR'01*, pages 334–342.