Browsing-based User Language Models for Information Retrieval

Fernando Diaz and James Allan
Center for Intelligence Information Retrieval
Department of Computer Science
University of Massachusetts
Amherst, Massachusetts

{fdiaz, allan}@cs.umass.edu

ABSTRACT

Traditional information retrieval systems have ignored the potential improvement in precision provided by personalization. We present a study of the behavior and evaluation of personalized information retrieval systems. We describe the construction of a collection of user web browsing data for application in retrieval evaluation. Several novel techniques for personalizing retrieval are presented and evaluated. Although performance is mixed, results point to the need to develop other algorithms within this evaluation framework.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Search Process

Keywords

Personalization, Information Retrieval, World Wide Web

1. INTRODUCTION

Traditionally, information retrieval systems have dealt with the task of ranking documents for a given query. While these systems have become quite successful, recent attempts have been made to increase precision by introducing concepts such as passage retrieval, hypertext retrieval, and question answering. Our work contributes to these tasks by presenting the problem of formally incorporating user models into information retrieval. We hypothesize that this perspective should allow for the development of algorithms which increase precision by disambiguating the information need and personalizing retrieval.

Our research approaches two aspects of user modeling. First, we use traditional information retrieval techniques to investigate the dynamics of user behavior. While assumptions such as user interest predictability make sense intuitively, it is not clear whether information retrieval techniques can capture or exploit this behavior. Second, we address the issue of designing and evaluating personalized information retrieval algorithms. While current test collections provide a foundation for evaluating traditional systems, none incorporate rich user histories. Meanwhile, personalization collections mainly deal with systems for providing recommedations as opposed to retreival. We present an approach to capturing and annotating user browsing histories with the end of evaluating personalized retrieval systems.

This paper will begin with a description and analysis of our dataset and follow on to use our insights to develop and evaluate personalized information retrieval algorithms. After placing our research in context (Section 2), we describe the monitoring and collection of user browsing behavior in Section 3. Section 4 the language modeling framework which we use in Section 5 to build user language models for analyzing the dynamics of user browsing behavior. We then place these results in the context of information retrieval by first describing the annotation of the test collection (Section 6). This collection is then used in Section 7 to evaluate the performance of several novel information retrieval tasks.

2. RELATED WORK

Although there are many studies of user browsing behavior, very few studies have taken an information retrieval perspective. Work by Jansen *et al* presents a thorough discussion of gross web searching behavior [7]. While describing aggregate tendencies, the study lacks a direct application to personalized information retrieval system design. Chi *et al* [3] and Huberman *et al* [6] have conducted research on information-seeking behavior in a web environment. Again, these studies point to broad regularities in user behavior but fail to present results which are useful for designing retrieval algorithms.

Attempts to introduce user modeling into information retrieval have been smaller in scale, focusing on the software design aspects of the problem. The Watson [2] and Letizia [10] systems use term-weighting for constructing models of user interests. Although similar to our approach, these systems are more grounded in capturing the immediate context of the user rather than the longer-term topical interest.

There has been research in information retrieval regarding user modeling *within* a search session mostly in the guise of interactive information retrieval and relevance feedback [1, 9]. These systems model a user's changing information need within a search session. Our research leverages longer term models and implicit feedback from a user's browsing patterns *between* sessions.

Another relevant piece of work deals with content-based pre-caching of web pages [5]. Pre-caching seeks to predict the most likely hyperlink a user will request given a brief history of content. This procedure results in lower browsing latency and a better user experience. Our work applies a similar idea to the context of

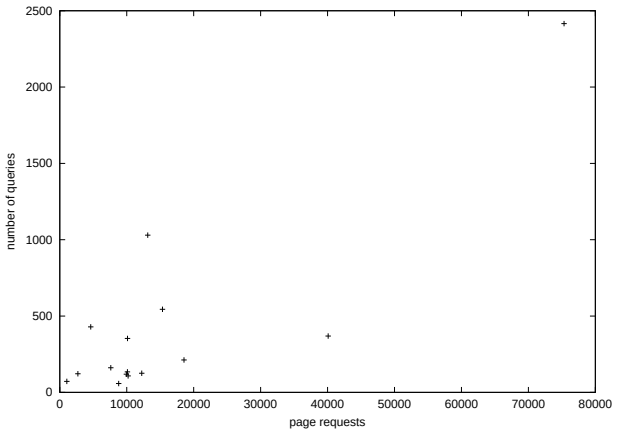

Figure 1: Requests vs. Searches (outliers at 40,000 and 80,000 page requests).

$D(U_1\|\|C)$	$D(U_1\|\|U_2)$	$D(U_2\|\|U_1)$
cnn	cnn	salon
com	wfmu	tivo
video	playlist	umpire
wfmu	video	redoctane
playlist	sci	fort
news	time	river
search	reply	coed
sci	edition	ddr
music	asia	game
tech	space	forum
entertain	denote	pad
edition	score	playstation
time	instyle	netscape
reply	scoreboard	letter
page	eht	comic

Figure 2: Qualitative comparison between user and collection language models.

information retrieval.

3. MONITORING SETUP

We are interested in gathering data from which to construct meaningful user models for information retrieval. We capture this information by monitoring a set of users' web browsing habits. Over the course of nine months, 24 individuals from our department redirected their web browsers to a proxy server which transparently intercepted requests. The proxy server cached all HTTP requests and content. Only HTML content was used for this study although PostScript and PDF documents were also saved.

User page requests were highly skewed with the most active participant viewing four times as many pages as most of the others. Figure 1 plots the the number of page requests against the the number of queries. For each user, we determine a querying rate by inspecting the how often they submitted a query to a search engine. Experiments are marked in cases where outlying users were removed.

4. LANGUAGE MODELING APPROACH

The language modeling framework will be used as a tool for studying and incorporating user information. The assumption behind language modeling is that a document or group of documents can be seen as having been generated by some probabilistic process. Using the actual text as a sample from this distribution, we can estimate the generating model. The language modeling framework provides a theoretically sound and empirically successful methodology for studying information retrieval [4].

Our language models are constructed by using a maximum likelihood estimate smoothed with a general English language model,

$$p(w) = \lambda \frac{\sum_{d \in S} \#(w,d)}{\sum_{d \in S} len(d)} + (1-\lambda) \frac{\sum_d \#(w,d)}{\sum_d len(d)} \quad (1)$$

where S is some arbitrary set of documents from the collection and $\#(w,d)$ refers to the number of times word w occurred in document d; the interpolation parameter, λ, is set to .9 in our experiments. More sophisticated smoothing methods exist but were not used in this study [14]. Our composite collection is the combination of all documents viewed by all users. This collection is used to construct the general english or composite language model.

We will be using two methods for comparing documents and language models. When a single document is compared to a language model, we calculate the probability of the language model having generated the document,

$$p(d|M) = \prod_{w \in d} p(w|M). \quad (2)$$

When dealing with groups of documents, we construct a second language model according to equation 1 and compare the two models. A common method of comparing two language models is to compute the Kullback-Leibler divergence between the distributions. The KL divergence is defined as

$$D(p||q) = \sum_w p(w) \log \frac{p(w)}{q(w)} \quad (3)$$

While the KL-divergence generates a single number for the dissimilarity between language models, individual words can be ranked according to the contribution to this score. We will use this method for qualitative analysis between language models.

5. USER LANGUAGE MODELS

We begin by analyzing user browsing behavior in the language modeling context. This initial analysis attempts to study how predictable user behavior can be in general. In Section 6, we will analyze the predictability of information needs specifically.

5.1 Static User Models

Our first user language model is constructed from the documents in a user's browsing history. We can qualitatively compare these static models to arbitrary language models. Figure 2 presents the qualitative description of a particular user. In the first column, we present the difference between this user and the composite language model. Here, the larger differences arise in terms for news (e.g. "cnn", "news") and music (e.g. "wfmu", "playlist", "music"). These characteristic terms persist when we compare this user to another user in the second column. However, when we look at this second user with respect to the first, it is apparent that the second user is interested in electronics (e.g. "tivo", "playstation", "redoctane") and softball (e.g. "umpire", "coed").

While this analysis provides some intuition that there is meaningful information contained in user models, we would like a quantitative measure of this behavior. In order to test its usefulness, we evaluate how well this model distinguishes between documents a user viewed and a random set of documents. This can be accomplished by first inspecting the probability of documents in the user history with respect to the user language model; similarly, the probability of a random history of documents can be computed with respect to the user model. Figure 3 shows a time series of probabilities for these two sets of documents for U_1 from Figure 2. In this case, the behavior of the user is quite unpredictable and indeed is comparable to a random history of documents.

While this individual gives us the impression that the language modeling framework does not capture the subtleties of locality, we would like to confirm this fact for all users. The cumulative probability distribution of scores over users is plotted in Figure 4. Here, we are interested in the point where a threshold can be established for distinguishing between empirical and random documents for a large proportion of documents. Unfortunately, this point appears to be quite high while capturing a small percentage of the empirical documents. This indicates that the likelihood of random and empirical documents are similar when using the static user model. We would like, then, to build user models which can better distinguish between these two streams.

5.2 Dynamic User Models

One possible way to improve predictability is to construct a dynamic model as opposed to a static model. That is, we can partition the stream of documents into subsets representing months and build models according to Equation 1. These models can then be compared to the static user language model in order to get a ranked list of distinguishing terms. Figure 5 depicts the change in interest for User 1; each column represents a month. Immediately noticeable are the higher ranks of the names of the months. Furthermore, we can tell from the model that this user began looking for apartments in early summer (second and third columns). Throughout the history, the shifting importance of several news topics can also be seen. For example, the prominence of articles about Israel in April of 2002 disappears toward the end of the summer when the issues of the Fall of 2002 begin to appear (Iraq, D.C. sniper).

As an alternative to partitioning the user history, a dynamic language model can be continually updated by using a sliding window of documents. In these experiments, we test the ability of a particular document to be predicted by a language model built from preceding documents. Using the methodology described in the previous section, we have plotted the cumulative probability distributions for these scores in Figure 6. Notice that both window sizes are comparable at discriminating between the random and empirical documents and are superior to the static user model. This indicates a strong tendency to continue browsing recently viewed topics but little global consistency.

6. BUILDING A TEST CORPUS

So far the relationship of these user histories to information retrieval has remained secondary. In order to mine potentially useful retrieval information, participants were asked to annotate parts of their history with relevance information. Specifically, we wanted the participants to define segments of their histories where they were searching for information. Session starts were automatically detected as any point where a user issued a query to a search engine. After detecting these query points, we asked users to mark the end of their search session and annotate all documents between the initial query and the last session document. Documents were marked as either:

1. Reformulation: for modified queries to the search engine
2. Relevant: for documents which satisfy the information need
3. Somewhat Relevant: for documents which partially satisfy the information need
4. Not Relevant: for documents which did not satisfy the information need at all
5. Browsing: for documents which were used to navigate toward relevant information
6. Not Related: for documents which were not at all related to the search

Figure 7 shows how session might be annotated. Only the documents within the session are annotated. Related documents not explicitly part of the search are left unmarked.

Using this data, we can gauge the success of current search technology. If a successful search session is defined as one where a relevant or semi-relevant document is encountered, then roughly 70% of searches terminated in success. Figure 8 displays the distribution of session lengths for both successful and unsuccessful sessions. Moreover, of these successful sessions, 36% required browsing beyond the initial results page while 85% of the unsuccessful sessions might be accounted for by a lack of browsing.

Since we have the relevance judgments for a set of queries for each user, we can also detect the predictability of search topics. Using the methodology from Section 5, Figure 9 displays the cumulative probability distributions of relevant documents with respect to the previous 5 and 50 documents. Here, the preceding documents do not appear to provide information about the subsequent search topics, indicating that information needs may be arising from sources outside of a browsing experience. That is, users usually seek topics they are not familiar with.

7. RETRIEVAL EXPERIMENTS

Although the user history apparently provided little information about the information needs with respect to a random model, we wanted to examine the usefulness of the history with respect to the documents in the session. One can imagine user information being used as a post-processing step to personalize results from a more general information retrieval system. In order to evaluate algorithms for performing such a task, systems were required to re-order search session documents so that relevant documents occurred first. Then, we performed traditional information retrieval evaluations on this subset of documents.

All systems compared are based on the language modeling framework. Our best case system assumes that we have access to the "true relevance model" defined by the language model built from the known relevant documents. The worst case system uses a relevance model constructed from random documents. The baseline system ranks the documents according to the original order they were viewed by the user.

7.1 Retrieval Algorithms

We investigate the performance of four ranking algorithms. The first two systems ignore user behavior while the second two incorporate user information.

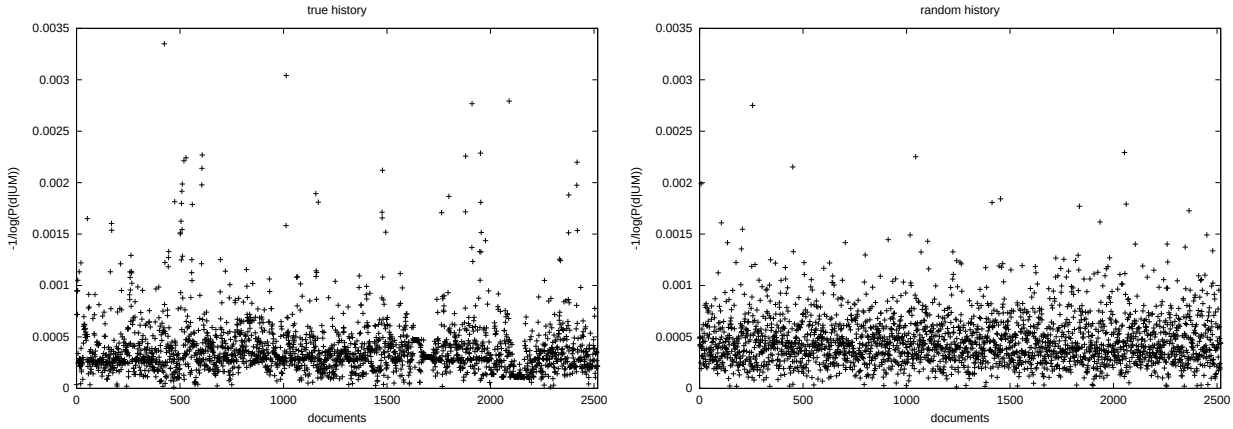

Figure 3: Locality of browsing given user history.

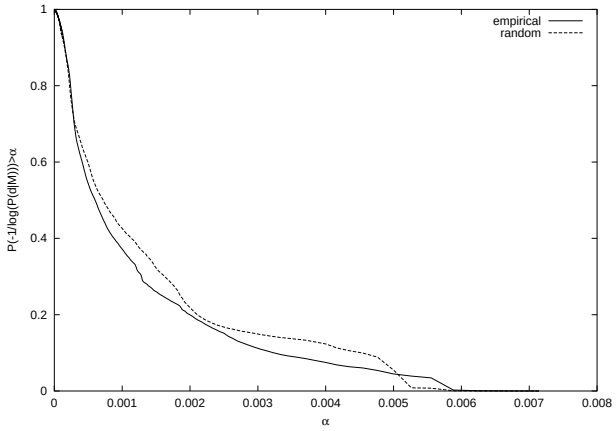

Figure 4: Cumulative probability distribution of scores for users (outliers removed)

April	May	June	July	August	September	October
reply	search	june	july	linux	hour	linux
video	plot	rent	rent	relate	news	snipe
powell	song	liquid	wfmu	august	september	i386
seltaeb	page	brazil	playlist	relevant	iraq	october
function	gnuplot	pledge	score	html	relate	gnome
time	bomb	samba	bedroom	perl	tribune	news
animal	similar	bedroom	listen	module	failure	slashdot
input	network	music	unknown	date	unite	reuter
speech	rock	apartment	mine	locale	minute	drudge
israel	mozilla	heat	music	file	time	moscow

Figure 5: Qualitative change in interest over time.

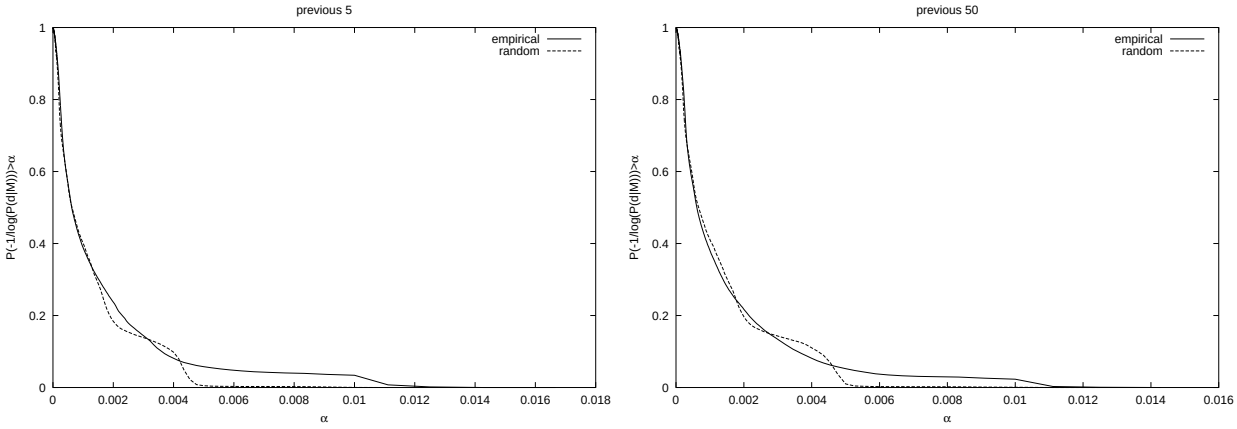

Figure 6: Locality of browsing given (a) a 5 document window and (b) a 50 document window (outliers removed).

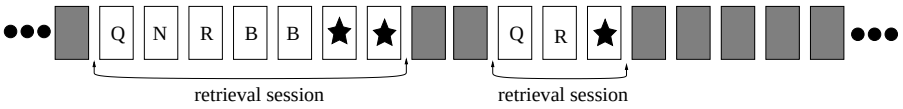

Figure 7: History annotation. Users were asked to mark the end of retrieval sessions started by points where queries were issued to search engines. The first document is always the results list from the initial query (Q). Other documents in the session may be relevant (star), somewhat relevant (S), not relevant (N), useful as for browsing to the relevant documents (B), or results of reformulations of the initial query (R).

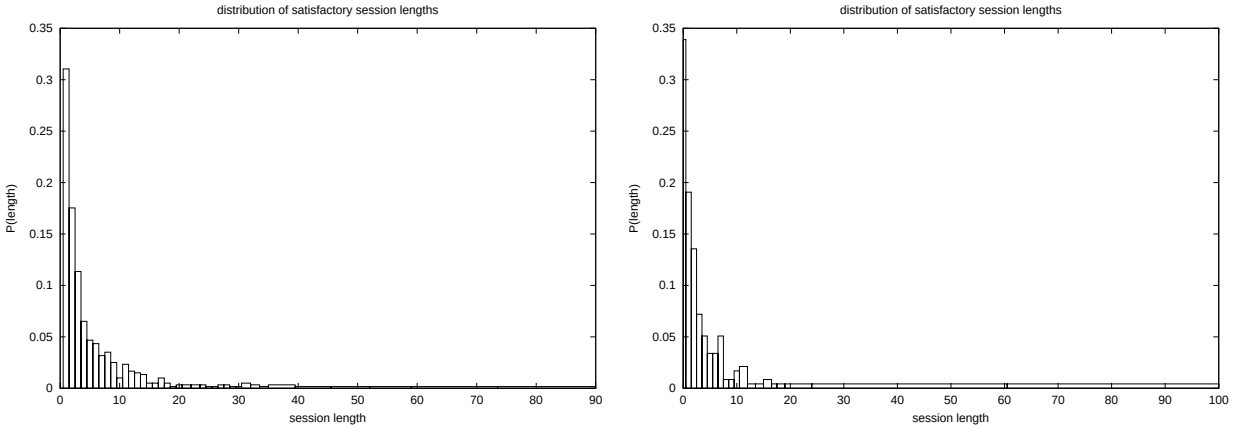

Figure 8: Distribution of session lengths.

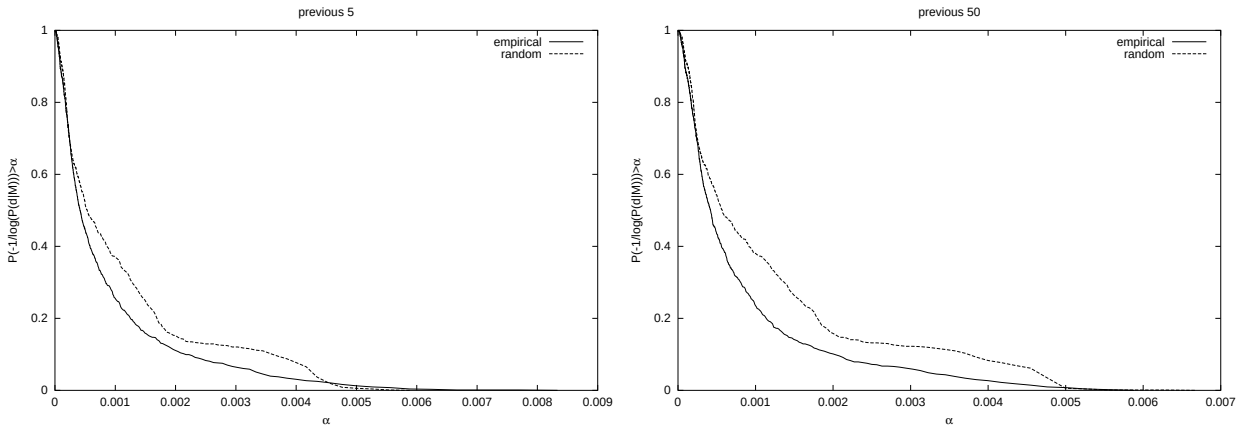

Figure 9: Distribution of scores with respect to the true relevance model and a random model.

7.1.1 Query Likelihood (ql)

We rank the documents in the session according to the likelihood of having generated the query. This is accomplished by constructing a language model for each document and ranking documents according to $P(Q|D)$.

7.1.2 Approximated Relevance Model (arm)

We can also attempt to approximate the true relevance model [8]. We present two methods of approximating this model. In both methods, we are using the results of an initial query likelihood retrieval to construct the approximated relevance model. In the first approximation (arm1), we build a new language model using session documents weighted by their query likelihood. In the second approximation (arm2), we build a new language model using the top documents returned from a query to the composite collection. The approximated relevance model is then used to rank the session documents according the KL-divergence between the document language model and approximated language model.

7.1.3 Recent Topic (p50)

At an extreme, we assume that the current information need can be represented by the most recent documents. In this evaluation, it seems reasonable that the documents in the session will all be relatively similar and that the the distinguishing ability of dynamic models would provide better ranking. In these experiments, we use a window size of 50.

7.1.4 Query Collaboration (qc)

Several retrieval systems have found it advantageous to incorporate the relevance judgments of historic queries [11, 12]. Such systems improve performance by either relying upon the predictability of individual user querying behavior or aggregate querying behavior. In this spirit, we developed a system to use the feedback from other search sessions. Specifically, for each query, we allow the system to inspect the relevant documents of all other sessions. These subsets are used to build a database of topic language models.

$$p_{S_i}(w) = \lambda \frac{\sum_{d \in S_i} \#(w,d)}{\sum_{d \in S_i} len(d)} + (1-\lambda) \frac{\sum_d \#(w,d)}{\sum_d len(d)} \quad (4)$$

where S_i is the subset of relevant documents from ith session. At query time, then, the cached topic language model most likely to have produced the query is used as the model for ranking of session documents. Such a system harnesses individually local as well as globally popular topic models, unifying the disparate approaches found in previous literature.

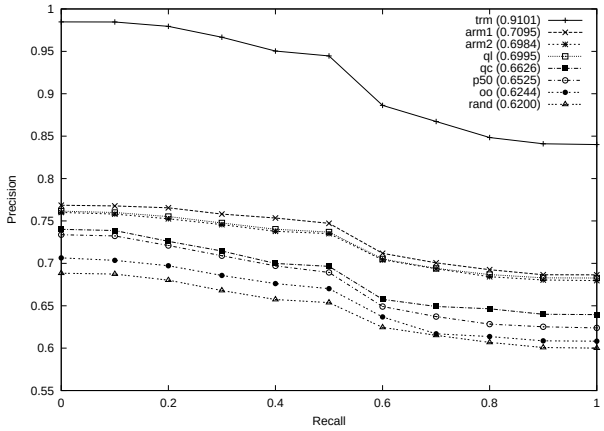

Figure 10: Precision-Recall curves for re-ranking task using the true relevance model (trm), approximated relevance model (arm1, arm2), query likelihood (ql), query collaboration (qc), recent topic (p50), original order (oo), and random model (rand).

7.2 Results

Only successful sessions which included more than 3 documents were considered. Query text was automatically extracted from the first submission to the search engine; reformulations were not included in the query text. This resulted in a set of 201 query sessions. Figure 10 shows the results for our algorithms. Important to note is the fact that the original order is almost as poor as the random language model. Part of this is due to the nature of the search process which may only succeed at the end of a session. The true relevance model expectedly performed quite well while the traditional approaches (arm1, arm1, and ql) perform better than the new algorithms. That the new techniques perform poorly confirms the unpredictability of the information needs since both rely upon topic model predictability.

In order to test the robustness of the algorithms, we also attempted the same experiment with the original set of documents

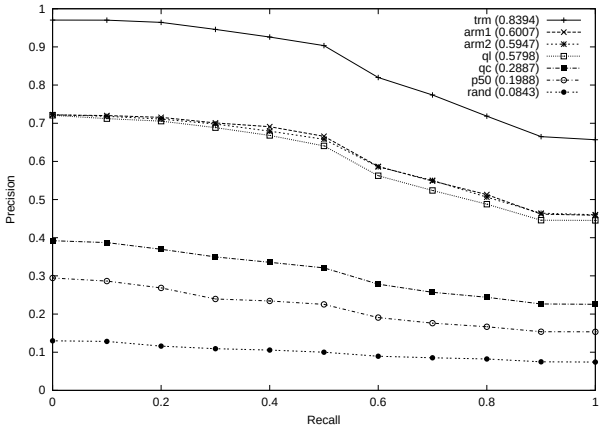

Figure 11: Precision-Recall curves for inflated re-ranking task. Note that the precision scale is different from Figure 10.

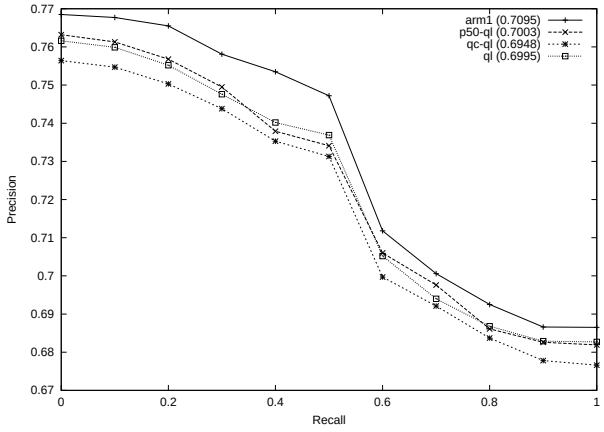

Figure 12: Precision-Recall curves for original re-ranking task using interpolated models

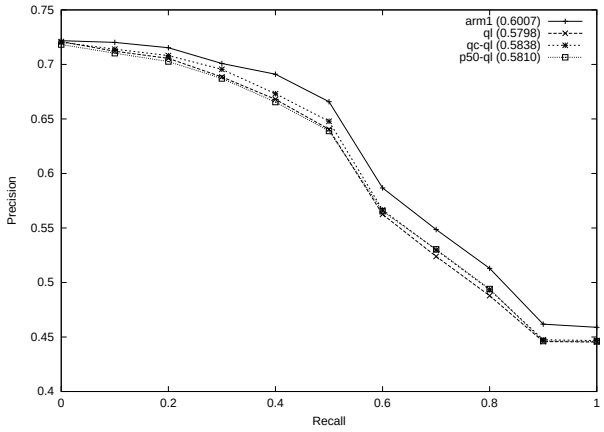

Figure 13: Precision-Recall curves for inflated re-ranking task using interpolated models

augmented by random documents selected from the composite collection. Results are displayed in Figure 11. All systems performed worse on this task although the new systems were hurt the most. However, the newer systems abandon more of the original query than the traditional systems. For example, qc assumes the existence of topical information in the database of other sessions; p50 assumes the existence of topical information in the previous 50 documents. However, the former may not be true while the latter was shown to be false in Section 6. The traditional systems rank based upon models close to the original query either explicitly (ql) or implicitly (arm1, arm2) because of the existence of relevant documents.

The new retrieval algorithms described all operate by replacing the query with a new model. As an alternative, we experimented with incorporating p50 and qc into our more successful methods. One of the problems with p50 is that potentially irrelevant information is included in the language model. This is evidenced by the lack of information need predictability. In order to tease out the relevant documents, we chose the 10 documents from the preceding 50 which were most likely to have generated the query. These documents are then used to construct a new language model which is combined with the original query according to,

$$p(w) \quad = \quad \gamma p_{r10}(w) + (1-\gamma)\frac{\sum_{w \in q} \#(w,q)}{\sum_d \#(w,q)} \qquad (5)$$

where p_{r10} is the language model built from the 10 most relevant documents from the recent history and γ is an interpolation parameter set to .1 for our experiments. The qc language model is incorporated in a similar way except that we ignore the initial ranking and pruning of the document set,

$$p(w) \quad = \quad \gamma p_{qc}(w) + (1-\gamma)\frac{\sum_{w \in q} \#(w,q)}{\sum_d \#(w,q)} \qquad (6)$$

where γ is again set to .1. Figures 12 and 13 present the results of combining these models. While small, the improvements over the ql method alone provide some intuition as to the benefit provided by each model. In the case of re-ranking the original session documents (Figure 12), p50 improves ql at low recall because the information provided by this model is helpful for personalizing a set of already similar documents. In the case of ranking the inflated sessions (Figure 13), qc improves ql because the information provided by this model is helpful for enriching the topical model. It is anticipated that as the size of the collection of sessions increases, this effect will be more substantial given repeated querying behavior [13].

8. FUTURE WORK AND CONCLUSION

While the study presented here indicates that information needs are unpredictable, we believe that useful information exists in browsing history. As such, we are further refining our approach the analysis, construction, and evaluation of personalized information retrieval systems.

Initially, we are considering the addition of other sources of information. We conjecture that information needs may become more predictable if linguistic data such as open documents and email are included in dynamic user models.

In addition to considering more linguistic information, we are also interested in expanding the sets of sessions for our qc algorithm. Since other research points to highly redundant querying behavior, incorporating what models of what other users found relevant may improve the performance of qc much more.

We have presented initial attempts to combine an understanding of user browsing and retrieval behavior with information retrieval. The results presented point should help in the development of more

sophisticated algorithms than what has traditionally been applied on small scales. Furthermore, we have described a methodology for constructing and annotating a set of user histories for the evaluation personalized information retrieval systems.

9. ACKNOWLEDGMENTS

The authors would like to thank Haizheng Zhang for help in setting up the web proxy. This work was supported in part by the Center for Intelligent Information Retrieval and NSF grant #IIS-9907018. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor.

10. REFERENCES

[1] N. J. Belkin. Interaction with texts: Information retrieval as information-seeking behavior. In *Information Retrieval*, pages 55–66, 1993.

[2] J. Budzik and K. Hammond. Watson: Anticipating and contextualizing information needs. In *62nd Annual Meeting of the American Society for Information Science*, Medford, NJ, 1999.

[3] E. H. Chi, P. Pirolli, and J. Pitkow. The scent of a site: a system for analyzing and predicting information scent, usage, and usability of a web site. In *Proceedings of the CHI 2000 conference on Human factors in computing systems*, pages 161–168. ACM Press, 2000.

[4] B. Croft. *Language Modeling for Information Retrieval*. Kluwer Academic Publishers, Boston, Massachusetts, 2003.

[5] B. D. Davison. Predicting web actions from html content. In *Proceedings of the The Thirteenth ACM Conference on Hypertext and Hypermedia*, pages 159–168, College Park, MD, 2002.

[6] B. A. Huberman, P. L. T. Pirolli, J. E. Pitkow, and R. M. Lukose. Strong regularities in World Wide Web surfing. *Science*, 280(5360):95–97, 1998.

[7] B. J. Jansen, A. Spink, J. Bateman, and T. Saracevic. Real life information retrieval: A study of user queries on the web. *ACM SIGIR Forum*, 32(1):5–17, 1998.

[8] V. Lavrenko and W. B. Croft. Relevance-based language models. In *Research and Development in Information Retrieval*, pages 120–127, 2001.

[9] A. Leuski and J. Allan. Interactive information retrieval using clustering and spatial proximity. *submitted to User Modeling and User-Adapted Interaction Journal*, 2002.

[10] H. Lieberman. Letizia: An agent that assists web browsing. In C. S. Mellish, editor, *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)*, pages 924–929, Montreal, Quebec, Canada, 1995. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA.

[11] Z. Lu and K. S. McKinley. *Advances in Information Retrieval: Recent Research from the Center for Intelligent Information Retrieval*, chapter The Effect Of Collection Organization And Query Locality On Information Retrieval System Performance And Design. Kluwer Academic Publishers, 2000.

[12] E. M. Voorhees, N. K. Gupta, and B. Johnson-Laird. Learning collection fusion strategies. In *Proceedings of the 18th International Conference on Research and Development in Information Retrieval*, Seattle, WA, 1995.

[13] Y. Xie and D. O'Hallaron. Locality in search engine queries and its implications for caching. In *IEEE Infocom 2002*, pages 1238–1247, New York, June 2002. IEEE.

[14] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Research and Development in Information Retrieval*, pages 334–342, 2001.