Selective User Interaction

Giridhar Kumaran and James Allan
Department of Computer Science
University of Massachusetts Amherst
Amherst, MA 01002, USA
giridhar,allan@cs.umass.edu

Categories and Subject Descriptors: H.3.3 Information Search and Retrieval[query formulation, search process]

General Terms: Experimentation, Human Factors

Keywords: User Interaction, Query Expansion, Query Relaxation, Query Potential

1. INTRODUCTION

Query expansion [12] refers to the process of including related terms in the original query to produce *expanded* queries, while query relaxation [8] refers to the dropping or down-weighting of terms from the original query to produce *sub-queries*. The automatic versions of both query expansion (AQE) and query relaxation (AQR) are known to fail in a large fraction of queries, and overall (average) improvements in performance can be attributed to high gains on a smaller fraction [7].

The potential to address the mistakes made by automatic techniques by involving the user [6] motivates interactive versions of these techniques (IQE, IQR). Previous research has shown that involving users in selection [4, 5, 10, 1] or rejection of terms or sets of terms [8] suggested by an automatic method has the potential to further improve performance. However, the same problems that plague automatic techniques are prevalent in interactive techniques: i.e. user interaction has the potential to lead to improvements only for a subset of queries. Further, a second problem has generally been ignored: frequently none of the options selected by the automatic procedures and presented to the user are any better than the original query. In this paper we develop and present procedures for determining when to interact with a user to obtain explicit feedback in the IQR and IQE settings. We show that by using these procedures we can avoid interaction for almost 40% of TREC queries without compromising significant improvements over the baseline. We also develop procedures to rank queries by their potential for improvement through user interaction, enabling systems to interact with users working under time and cognitive load constraints.

CIKM'07, November 6–8, 2007, Lisboa, Portugal.

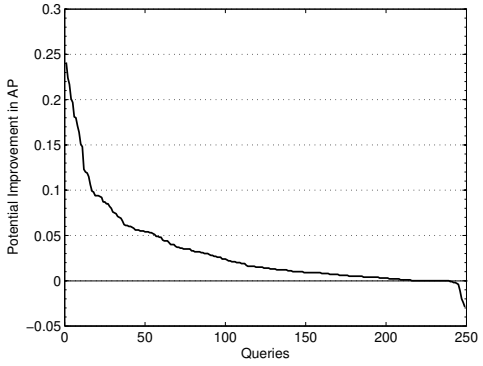

Figure 1: Query Relaxation: The utility of interaction on a per-query basis. Values less than zero (to the right) indicate that none of the sub-queries presented to the user were better than the baseline query

The motivation for this paper is best summarized through Figure 1. It shows the distribution of the *absolute* potential improvements in mean average precision (MAP) due to IQR for 249 queries in the TREC Robust 2004 collection. If one were to consider a minimum improvement of 0.025 to be worth interacting to achieve, then we can see that user interaction for close to 150 queries is unnecessary. Identical trends are observed for IQE as well. The overall improvements in MAP (from 0.235 to 0.332, and 0.261 to 0.341 respectively) mask the minuscule improvements contributed by these queries.

In this paper we seek to address the following questions, *Given a long query, is it possible to infer the potential utility of invoking user interaction to select a relaxed version of the same query?* and *Given a short query, is it possible to infer the potential utility of invoking user interaction to select a better set of expansion terms?*

2. EXPERIMENTS

We used version 2.3.2 of the Indri search engine, developed as part of the Lemur[1] project. The pseudo-relevance feedback mechanism we used was based on relevance models [9]. We reprised our earlier maximum spanning tree based algorithm [8] to rank sub-queries/expanded queries. We used

[1] `http://www.lemurproject.org`

the TREC Robust 2004, Robust 2005, TREC 5 ad-hoc and HARD 2003 document collections for our experiments. All collections were stemmed using the Krovetz stemmer provided as part of Indri. We also used a manually-created stoplist of twenty terms (*a, an, and, are, at, as, be, for, in, is, it, of, on, or, that, the, to, was, with* and *what*). 249 queries from the TREC Robust 2004 track were analyzed to determine and fine tune the procedure we developed to determine the utility of interaction. The remaining 150 queries, 50 each from the three remaining tracks, were used to test the effectiveness of our interaction-utility determining procedure. We measured performance using mean average precision (MAP)and geometric mean average precision (GMAP).

There is a large body of previous and related work on procedures to determine the quality of queries [13, 3, 2]. The goal of that work was to predict in advance if a query will result in acceptable values of precision, and take appropriate steps if the query was predicted to fail (have a low AP). The procedures were thus tuned to accurately predict MAP. Our goal is different. We wish to determine if an interaction mechanism will lead to an improvement in MAP. From the perspective of a user, expending interaction effort to improve precision from 0.1 to 0.11 is of the same utility as improving precision from 0.8 to 0.81 i.e. little utility. Hence we tuned our procedure to target improvements in MAP, and not just MAP values themselves.

2.1 Predictive features

Our investigation of potential features for predicting improvement was guided by the following hypotheses about potentially good question sets for interaction. By *question sets*, we mean the set of top ten sub-queries or expanded queries presented to the user.

1. When the original query is very long, a large number of extraneous terms are present that hinder retrieval instead of supporting it[2]. Thus, question sets that have low average length, or are derived from shorter queries, are potentially better
2. The average score [8] of the question sets will be high, indicating a very focused set of queries
3. The scores of the sub-queries/expanded queries in the question sets will be diverse, indicating that they cover different aspects of the query.

For each query, we started with the top ten sub-queries / expanded queries ranked by the selection procedure we developed in [8]. We used the scores assigned to them by the selection procedure to investigate several features based on measures of central tendency, measures of dispersion, and measures involving query lengths. In this paper we report only those features that had a high coefficient of correlation (γ) with MAP. Table 1 provides a list of the top features we found correlating with potential improvements in AP in the case of IQR and IQE.

The feature with the highest correlation in IQR was original query length (QL). The negative value indicates that high values of initial query length translate to low-quality sub-queries, while lower values of initial query length are predictive of high-quality sub-queries. This is intuitive as identifying all the concepts in longer queries is more difficult. Longer queries also tend to induce more errors into the sub-query ranking procedure. The feature with the second highest correlation was a dimensionless quantity, coefficient of variation, $CV = \frac{s_x}{\bar{x}}$, where s_x is the standard deviation of a set of samples x_i, and $\bar{x}$ its mean. CV can be considered as a measure of the scatter of a set of values. The positive correlation indicates that question sets that have high dispersal are more likely to contain sub-queries that lead to improvements in AP. This is consistent with our hypothesis that question sets with varied sub-queries are more likely to cover concepts the user is interested in. Interestingly, the coefficient of correlation between QL and CV is -0.361. Volatility log change is the standard deviation of the natural base logarithms of the differences of successive ordered values of a set X i.e. $VC = \sigma(Y)$ where $y_i = \log \frac{x_{i+1}}{x_i}$. Since the volatility measure had very high correlation with CV, we chose to use only CV as a predictive measure for IQE.

[2]Identifying and selectively weighting such terms is a continuing challenge

Feature	γ
Interactive Query Relaxation	
Original query length	-0.305
Coeff. of Variation	0.245
Mean score	-0.239
Median score	-0.236
Interactive Query Expansion	
Coeff. of Variation	0.267
Volatility log change	0.171

Table 1: Features with the highest correlation coefficient with respect to potential improvement in AP

2.2 Selecting thresholds

Using training instances we learned a simple decision tree thresholded on feature values to determine when to interact with a user. Table 2 reports the change in potentially achievable MAP as well as the percentage of queries requiring user interaction for IQR when simultaneous threshold sweeps on both features, QL and CV, were performed. Every MAP value in the table is a statistically significant improvement over the baseline of 0.235.

It is apparent from the table that a wide selection is available for determining appropriate thresholds for the two features. We chose values of 16 for QL, and 2 for CV (see boxed number in Table 2). For the training set, it meant obviating interaction for 97 i.e. (1.0-0.61)∗249 queries in lieu of a 2 % reduction in potential MAP improvement .

Table 3 reports the change in potentially achievable MAP and the number of queries requiring user interaction as a threshold-sweep is performed on CV in the case of IQE. The transition to non-significant improvements over the baseline as the threshold is increased shows the limit to which we can *avoid* user interaction without impacting performance seriously. We chose a CV value of 6 as the final threshold.

3. RESULTS

In Table 4 we provide an overview of results for simulated IQR when the system makes a decision to either interact with the user or go with the baseline query. We can see that when selective interaction was performed there was an

		Coefficient of Variation Threshold							
		1	2	3	4	5	6	7	8
	15	0.261, 57	0.260, 56	0.259, 53	0.258, 50	0.258, 46	0.257, 43	0.256, 40	0.254, 37
Query	16	0.262, 64	0.262, 61	0.260, 57	0.259, 53	0.258, 49	0.257, 45	0.256, 42	0.254, 38
Length	17	0.263, 68	0.262, 65	0.260, 59	0.259, 55	0.259, 50	0.257, 45	0.256, 42	0.254, 38
Threshold	18	0.264, 73	0.263, 69	0.261, 63	0.260, 57	0.259, 50	0.257, 45	0.256, 42	0.255, 38

Table 2: Query Relaxation: Effect on potential improvement in MAP due to simultaneously varying QL and CV thresholds. The numbers provided are <MAP,%queries requiring interaction> tuples. For example, to potentially achieve a MAP of 0.264 (last row, first column), we need to interact with the user for 73% of the test queries. The baseline was 0.235. Statistical significance tests were performed using the Wilcoxon signed-rank test, with α set to 0.05.

Coefficient of Variation Threshold									
1	2	3	4	5	6	7	8	9	10
0.289, 100	0.289, 100	0.289, 100	0.289, 98	0.288, 93	0.286, 81	0.282, 61	*0.269*, 14	*0.266*, 6	*0.263*, 2

Table 3: Query Expansion: Effect on potential improvement in MAP at various CV thresholds. The numbers provided are <MAP,%queries requiring interaction> tuples. An italicized score implies that it was not a statistically significant improvement over the baseline MAP of 0.261

	Robust 2005	TREC 5	HARD 2003
Baseline	0.160	0.142	0.227
Upper Bound	0.283	0.217	0.351
Auto Select	0.162	0.122	0.223
User Select	0.190	0.158	0.267
Thresholded Select	0.180	0.153	0.253
% drop in MAP	5.5	3.1	5.2
% queries dropped	42	40	44

Table 4: Final results for query relaxation. The reported values are those of MAP

	Robust 2005	TREC 5	HARD 2003
Baseline	0.239	0.159	0.315
Upper Bound	0.305	0.210	0.371
Auto Select	0.244	0.162	0.319
User Select	0.266	0.170	0.333
Thresholded Select	0.260	0.165	0.325
% drop in MAP	2.2	2.9	2.4
% queries dropped	22	32	52

Table 5: Final results for query expansion. The reported values are those of MAP

average drop of 40% in the number of queries the user had to interact with, leading to an average drop in performance of 4.6%. In spite of the reduction, the final MAP was significantly better than the baseline (Wilcoxon text,α=0.05). However, in the case of Robust 2005 and HARD 2003, there was a significant drop in performance from what would have been achieved if the user interacted with all the queries ('User Select'). For a user with only enough time to interact for 60% (or *not* interact with 40%) of the queries the significant improvement over the baseline is still worth it.

The results for our simulated IQE experiments are given in Table 5. Again, we observed statistically significant improvements over the baseline for all three collections. The greatest reduction in the number of queries requiring interaction was for HARD 2003. However the MAP achieved by our system was statistically less than that potentially achieved by interacting with all queries. As mentioned before, we believe the impact of this result is subjective.

We now extend the procedures we have developed for selective user interaction to the scenario where a user presents the system with a set of queries along with a condition that she is willing to only interact say, for x% of the queries. Such situations are not impossible to imagine as users frequently have constraints on the time and effort they are willing to spare. To maximize the benefit from user interaction, it is apt for the system to determine the x% of queries that would have most potential for improvement. The trends in Tables 2 and 3 indicate that higher values of potential improvements in AP correlate with higher values of CV. Guided by this observation, we sorted the question sets for each query in the descending order of CV values, and presented them to the simulated user.

Figure 2 provides an overview of the performance on Robust 2005 when the user accedes to interact for 10%, 20%, 30% and so on of the query set. The lowest curve shows the gradual improvements with increased user interaction when query subsets are chosen at random for interaction. The highest curve tracks the improvement when the system makes the best choice (highest potential improvement in AP) on queries for interaction each time. In between the two is the curve that conveys the effect of presenting the question sets in descending order of CV. While the potential for improvement does not rise as rapidly as in the upper bound case, it clearly is much better than presenting the user with queries in random order. The discrepancy in correspondence between the MAP at 60% interaction in the graph and the value reported in the table is because the latter's ordering of queries involves the second feature QL too. For the same user with time to spare for 60% of the queries, we can observe that using CV-based selection helps obtain better performance with the *same effort*, when compared to randomly selecting queries.

Figure 3 shows the potential gains obtained by increased user interaction through IQE on the Robust 2005 corpus. We notice that in the ideal case upper bound performance can be achieved by interacting with only 50% of the queries.

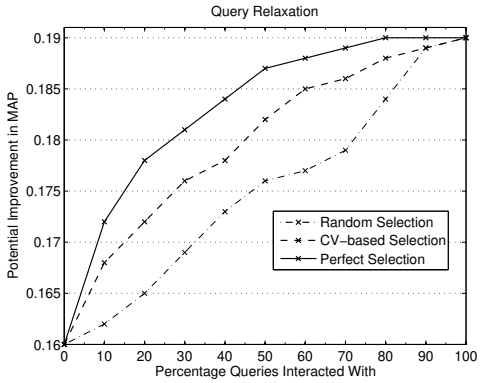

Figure 2: Trajectories of potential improvements in MAP using various question-selection techniques for Robust 2005 in IQR

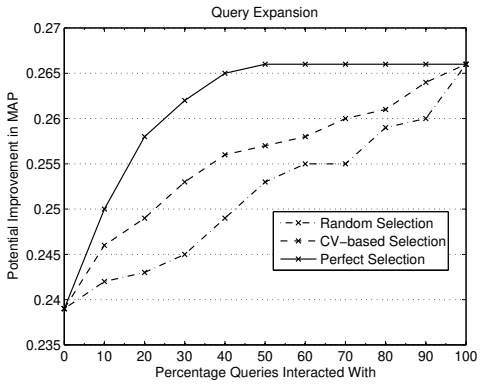

Figure 3: Trajectories of potential improvements in MAP using various question-selection techniques for Robust 2005 in IQE

In other words there was no utility in interaction for 50% of the queries. This explains the occasional 'flattening' of the CV-based selection and Random selection curves. The lower portion of the CV-based selection curve has a higher slope than the upper portion. This indicates that the selection process had done a good job of presenting queries with higher potential ahead of those with less.

4. CONCLUSIONS AND FUTURE WORK

We have discussed an important problem concerning interactive information retrieval systems. While user interaction is a promising way to improve retrieval effectiveness, its efficiency needs to be considered too. Inefficient interactive systems that force a user to interact on every instance can cause disenchantment. We have shown that it is possible to predict the utility of interaction with reasonable accuracy, and use it without compromising much on effectiveness. The use of a single feature measuring scatter for both interaction mechanisms implies that interaction mechanisms that provide a wide range of choices have more utility. In other words, showing the user the different parts of the search space her query could lead her to is advantageous.

Shen and Zhai [11] presented work whose motivation is similar to ours. They performed simulated user studies for interaction involving document-level feedback, with the goal of developing procedures that chose the best documents from a pool to present to the user for feedback. The procedures they developed for and results from such *active feedback* showed that showing users a diverse set of documents was most effective. However unlike our work on query reformulation, they did not extend theirs to determine when to interact with the user, or how to handle a user with time and cognitive load constraints.

Some extensions to our work include working with multiple interaction mechanisms and learning to select the most appropriate one based on a number of factors. Determining the optimal number of options to present to a user warrants further investigation too. Improving predictive accuracy by exploring new features is also an area of further interest.

Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval, in part by the Defense Advanced Research Projects Agency (DARPA) under contract number HR0011-06-C-0023, and in part by an award from Google. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor.

5. REFERENCES

[1] P. G. Anick and S. Tipirneni. The paraphrase search assistant: terminological feedback for iterative information seeking. In *SIGIR '99: 22nd ACM SIGIR Proceedings*, pages 153–159, 1999.

[2] D. Carmel, E. Yom-Tov, A. Darlow, and D. Pelleg. What makes a query difficult? In *SIGIR '06: 29th ACM SIGIR*, pages 390–397, 2006.

[3] S. Cronen-Townsend, Y. Zhou, and W. B. Croft. Predicting query performance. In *SIGIR '02: 25th ACM SIGIR Proceedings*, pages 299–306, 2002.

[4] D. Harman. Towards interactive query expansion. In *ACM SIGIR '98*, pages 321–331. ACM Press, 1988.

[5] D. Kelly, V. D. Dollu, and X. Fu. The loquacious user: a document-independent source of terms for query expansion. In *SIGIR '05: Proceedings of the 28th ACM SIGIR Conference*, pages 457–464, 2005.

[6] J. Koenemann and N. J. Belkin. A case for interaction: a study of interactive information retrieval behavior and effectiveness. In *CHI '96: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 205–212, 1996.

[7] G. Kumaran and J. Allan. Simple questions to improve pseudo-relevance feedback results. In *SIGIR '06: Proceedings of the 29th ACM SIGIR Conference*, pages 661–662, 2006.

[8] G. Kumaran and J. Allan. A case for shorter queries, and helping users create them. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 220–227, Rochester, NY, 2007.

[9] V. Lavrenko and W. B. Croft. Relevance based language models. In *SIGIR '01: 24th ACM SIGIR Conference Proceedings*, pages 120–127, 2001.

[10] M. Magennis and C. J. van Rijsbergen. The potential and actual effectiveness of interactive query expansion. In *SIGIR '97: Proceedings of the 20th ACM SIGIR*, pages 324–332, 1997.

[11] X. Shen and C. Zhai. Active feedback in ad hoc information retrieval. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 59–66, New York, NY, USA, 2005. ACM Press.

[12] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *ACM SIGIR '96*, pages 4–11. ACM Press, 1996.

[13] Y. Zhou and W. B. Croft. Ranking robustness: a novel framework to predict query performance. In *CIKM '06: Proceedings of the 15th ACM CIKM Conference*, pages 567–574, 2006.