UMass Genomics 2006: Query-Biased Pseudo Relevance Feedback

Mark D. Smucker

Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts Amherst

Abstract

Query-biased pseudo relevance feedback creates document representations for document feedback that aim to be more relevant to the user than using the entire document. Our submitted runs using query-biased feedback degraded performance compared to not using feedback. The cause of this degradation was the use of too many documents for feedback. Preliminary document retrieval experiments using fewer feedback documents found that query-biasing produced gains in the geometric mean average precision that non-biased feedback did not produce.

1 Introduction

The TREC Genomics track focuses on retrieval tasks typical of biomedical researchers [2]. This year the track switched from MEDLINE abstracts to a collection of biomedical research papers. The queries that are modeled by the Genomics track involve complex terminology that may be represented many ways in the documents [3].

We have found that a query-biased document-to-document similarity is better able to cluster relevant documents than a non-biased similarity [9]. Query-biased similarity places a window over each query term occurrence in a document. The words contained in these windows form the query-biased document. The query-biased relevant documents are more similar to each other than the original documents are.

We wanted to see how well query-biasing works for pseudo-relevance feedback, and we also felt it was well suited to address the problems with biomedical query terms. From the feedback documents, query-biased feedback uses only the words close to occurrences of the query terms. The nearby context of the query terms hopefully provides a model of the query terms that can match documents containing other notational varieties of the query terms but which have similar contexts. Since this approach is fully automatic, it it does not require any specialized database of synonyms and acronyms.

This year the Genomics track also looked at passage retrieval. The collection's research papers are considerably longer than the typical newswire article often used in other TREC collections. We investigated the well known technique of half-overlapping windows for passage retrieval. The track also looked at aspect retrieval, but we did not attempt to address aspect retrieval.

2 Methods

We used the Indri [11, 7] retrieval system for our experiments. We manually created our queries using the structured query language of Indri. We used the phrase and synonym operators as well as the standard bag-of-words operator, #combine. We did not manually add any words to the queries, but we did delete noise words. Our queries represent what might be expected from an experienced user of Indri's query language.

Lavrenko and Croft's [6] relevance models (RM) is an effective pseudo relevance feedback technique. The relevance model is a mixture of the top k documents taken from the results produced by running the initial query, Q. The relevance model M_R is calculated as:

$$P(w|M_R) = \sum_{i=1}^{k} P(D_i|Q)P(w|D_i)$$

where

$$P(D_i|Q) = \frac{P(Q|D_i)}{\sum_{j=1}^{k} P(Q|D_j)} \quad (1)$$

and $P(Q|D_i)$ is the Indri belief that document model D_i is relevant to the query Q. We combined the relevance model and the original query using Indri's

#weight operator to create the new query. The combination of the original query with the feedback model typically improves performance [1, 10, 8].

Query-biased (QB) pseudo relevance feedback works the same as relevance models, except rather than mixing maximum likelihood estimated (MLE) models of the feedback documents, it mixes query-biased document models. Our technique is similar to Xu and Croft's local context analysis [13] and to Lam-Adesina and Jones' method [5] but is simpler and designed to work within the language modeling approach to retrieval.

The query-biased document model is a MLE model of the document text that consists of all words within a certain distance W of all query terms in the document. For our experiments, we set W to 5. Thus the 5 preceding words, the query term, and the 5 words following a query term are used. This is the same procedure that we used for query-biased similarity [9] except that here we also counted stopwords.

The QB feedback models are mixed together using the weights of Equation 1 where the weights are based on the full document and not the query-biased document. We truncated the RM and QB feedback models to the 50 most probable terms.

We modified Indri to enable us to create these query-biased pseudo relevance feedback query models. Like RM, we combine the QB model with the original query to create the final QB query. We then used these queries with Indri's built in passage retrieval capabilities to create our three submitted runs: UMassCIIR1, UMassCIIR2, UMassCIIR1L.

UMassCIIR1 used 250 word half overlapping passages while UMassCIIR2 used 500 word half overlapping passages. The word count of an Indri passage includes stopwords. We post-processed the retrieval results to remove overlapping sections of passages. UMassCIIR1L is a "legalized" version of UMassCIIR1. The track supplied a set of legal passage spans, but UMassCIIR1 and UMassCIIR2 ignored these spans which excluded many of their results from the judging pool. UMassCIIR1L consists of the legal spans that were at least partially returned by the passages in UMassCIIR1. In addition, we deleted spans less than 750 characters from the output of UMassCIIR1L.

Indri supports the passage retrieval model of Wade and Allan [12] that uses Jelinek-Mercer smoothing and smooths the passage with its source document and then finally with the collection. We used the same parameter settings Wade and Allan found to work well giving both the passage and document a weight of 0.1 and the collection a weight of 0.8.

While our submitted passage retrieval runs used Jelinek-Mercer smoothing, we also report results for some preliminary document retrieval runs. For these runs and the initial retrieval for the feedback runs, we used Dirichlet prior smoothing [14]: $P(w|M_D) = \frac{D(w)+mP(w|C)}{|D|+m}$, where $P(w|C)$ is the MLE model of the collection, and m is the Dirichlet prior smoothing parameter.

Parameter	Value
Weight of original query model	0.5
Weight of pseudo feedback model	0.5
# feedback docs. (submitted runs)	1000
Max. terms in pseudo feedback model	50
# words in window, query-biased model	11
Words in passages (UMassCIIR[1,1L])	250
Words in passages (UMassCIIR2)	500
Jelinek-Mercer smoothing, passage λ	0.1
Jelinek-Mercer smoothing, document λ	0.1
Jelinek-Mercer smoothing, collection λ	0.8
Dirichlet smoothing, m	1500

Table 1: Retrieval parameters.

We used Indri's built-in ability to index the collection's 162,259 HTML documents. We stemmed all words with the Krovetz stemmer [4]. Our 424 stopwords consisted of the word stems that occurred in 50% or more of the documents.

Table 1 shows the retrieval parameters for all runs.

3 Results and Discussion

While we classified our runs as interactive, we did not create an interactive retrieval system. Our runs were "interactive" because we used manual queries and visually inspected the output of some of last year's queries and adjusted our parameters slightly.

Table 2 shows the results for our three runs. The larger passages for UMassCIIR2 appear to have helped its document and aspect performance while hurting its performance on passage retrieval. Our passage retrieval performance is disappointing. We suspect that one cause of this performance is that the relevance assessors selected much smaller passages than we had anticipated. UMassCIIR1 has an average passage size of 2269 characters while the relevant passages average 399.8 characters. UMassCIIR1L has an average passage size of 2294 and UMassCIIR2 averages 4429 characters.

The Genomics track is using a new passage retrieval metric adapted from the 2004 HARD track. A motivation for the use of character-based passage metrics in the 2004 HARD track was the 2003 HARD

Run	Mean Average Precision			Topics > Median		
	Document	Passage	Aspect	Doc.	Pass.	Asp.
UMassCIIR1	0.296	0.016	0.136	10	4	13
UMassCIIR2	0.332	0.010	0.176	10	3	17
UMassCIIR1L	0.265	0.018	0.114	7	5	10

Table 2: The arithmetic mean average precision for our three submitted runs and the number of topics for which the average precision was greater than the median of the 7 manual and 17 interactive runs submitted to the track.

track metric's sensitivity to passage size [12]. To test if the Genomics 2006 passage measure might also be sensitive to passage length, we took the UMassCIIR1 run, and repeatedly halved the passages. This halving process increased UMassCIIR1's passage MAP from 0.0164 to 0.0612 — a 273% improvement. The resulting average passage was 1.96 characters long. The Genomics passage metric also appears to be sensitive to passage size.

We thought that using the top 1000 documents for feedback would work well. The retrieval scores that are used to weight the feedback documents decrease rapidly and poorly ranked documents should contribute a negligible amount to the feedback model. In addition, we figured that the query-biased document models would stay sufficiently focused and relevant to the query that more document models could be used for feedback. We found that using 1000 documents was a significant mistake. We should have stuck with 50 or fewer feedback documents, which has worked well in the past.

Table 3 shows the document retrieval performance of 5 runs. Each run scores and retrieves 1000 documents directly and does not do any passage retrieval. This is in contrast to the submitted runs, which are passage retrieval runs for which document retrieval performance is also calculated. Our manual queries without pseudo feedback is our baseline. Each pseudo-relevance feedback run uses the baseline for their initial retrieval. Both RM and QB show similar performance compared to the baseline for arithmetic mean average precision (AMAP). When we use 1000 documents for feedback, both RM and QB do worse than the baseline, but when we use 10 documents, both do significantly better than the baseline. Our submitted runs performed comparably to the QB run with 1000 feedback documents.

Most interestingly, QB using 10 documents for feedback, has a statistically significant performance improvement of 15% in geometric mean average precision (GMAP) over the baseline while RM shows no performance improvement in GMAP. The geometric mean emphasizes the poorer performing topics. Thus, while RM and QB show similar gains in AMAP, QB is able to also significantly improve the performance of poorly performing topics.

These preliminary results appear to show that QB can do more with the same documents compared to RM. Both QB and RM use the same top 10 documents to compute a new model. The only difference between QB and RM is that QB computes different document models given the same source documents.

4 Conclusion

Using too many documents for relevance models and query-biased pseudo relevance feedback resulted in worse performance than not using feedback. Preliminary experiments using only 10 documents for query-biased feedback produced a 15% gain in the geometric mean average precision (GMAP) for document retrieval over the baseline. In contrast, relevance models using the same feedback documents did not increase GMAP. Our passage retrieval performance appears to have been hampered by choosing to return much larger passages than the relevance assessors wanted. Document and aspect retrieval performance were better with longer passages, but longer passages reduced passage retrieval performance.

Acknowledgments

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

References

[1] N. Abdul-Jaleel, J. Allan, W. B. Croft, F. Diaz, L. Larkey, X. Li, M. D. Smucker, and C. Wade.

Method	# Docs	AMAP	Pct.	ASL	GMAP	Pct.	ASL
Baseline		0.369			0.134		
RM	1000	0.347	-6%	0.301	0.099	-26%	0.132
QB	1000	0.343	-7%	0.147	0.101	-25%	0.055
RM	10	**0.402**	**9%**	**0.042**	0.133	-1%	0.952
QB	10	**0.405**	**10%**	**0.014**	**0.154**	**15%**	**0.005**

Table 3: Document retrieval results for the baseline, relevance models (RM), and query-biased (QB) pseudo relevance feedback. AMAP is the arithmetic mean average precision while GMAP is the geometric mean. # Docs is the number of top ranked documents used for pseudo relevance feedback. Pct. gives the percent change over the baseline. Bold results are statistically significant improvements over the baseline at an achieved significance level (ASL or p-value) of ≤ 0.05 as measured by a two-sided, paired, randomization test with 100,000 samples.

UMass at TREC 2004: Novelty and HARD. In E. M. Voorhees and L. P. Buckland, editors, *The Twelth Text REtrieval Conference (TREC 2003)*. Department of Commerce, National Institute of Standards and Technology, 2004.

[2] W. Hersh, A. Cohen, J. Yang, R. T. Bhupatiraju, P. Roberts, and M. Hearst. TREC 2005 genomics track overview. In *TREC 2005*. Department of Commerce, National Institute of Standards and Technology.

[3] X. Huang, M. Zhong, and L. Si. York University at TREC 2005: Genomics track. In *TREC 2005*. Department of Commerce, National Institute of Standards and Technology.

[4] R. Krovetz. Viewing morphology as an inference process. In *SIGIR '93*, pages 191–202. ACM Press, 1993.

[5] A. M. Lam-Adesina and G. J. F. Jones. Applying summarization techniques for term selection in relevance feedback. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 1–9, New York, NY, USA, 2001. ACM Press.

[6] V. Lavrenko and W. B. Croft. Relevance models in information retrieval. In W. B. Croft and J. Lafferty, editors, *Language Modeling for Information Retrieval*, volume 13 of *The Kluwer International Series on Information Retrieval*. Kluwer Academic Publishers, 2003.

[7] D. Metzler and W. B. Croft. Combining the language model and inference network approaches to retrieval. *IPM*, 40(5):735–750, 2004.

[8] D. Metzler, F. Diaz, T. Strohman, and W. B. Croft. UMass robust 2005 notebook: Using mixtures of relevance models for query expansion. In *TREC 2005 Notebook*, 2005.

[9] M. D. Smucker and J. Allan. Find-similar: Similarity browsing as a search tool. In *SIGIR '06*, pages 461–468. ACM Press, 2006.

[10] M. D. Smucker, D. Kulp, and J. Allan. Dirichlet mixtures for query estimation in information retrieval. Technical Report IR-445, CIIR, Department of Computer Science, University of Massachusetts Amherst, April 2005.

[11] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft. Indri: A language-model based search engine for complex queries (extended version). Technical Report IR-407, CIIR, Department of Computer Science, University of Massachusetts Amherst, 2005.

[12] C. Wade and J. Allan. Passage retrieval and evaluation. Technical Report IR-396, CIIR, Department of Computer Science, University of Massachusetts Amherst, 2005.

[13] J. Xu and W. B. Croft. Improving the effectiveness of information retrieval with local context analysis. *TOIS*, 18(1):79–112, 2000.

[14] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *SIGIR '01*, pages 334–342. ACM Press, 2001.