Query-Drift Prevention for Robust Query Expansion

Liron Zighelnic and Oren Kurland
Faculty of Industrial Engineering and Management
Technion — Israel institute of technology
Technion City, Haifa 32000
Israel
zliron@tx.technion.ac.il, kurland@ie.technion.ac.il

ABSTRACT

Pseudo-feedback-based automatic query expansion yields effective retrieval performance on average, but results in performance inferior to that of using the original query for many information needs. We address an important cause of this *robustness* issue, namely, the *query drift problem*, by *fusing* the results retrieved in response to the original query and to its expanded form. Our approach posts performance that is significantly better than that of retrieval based *only* on the original query and more robust than that of retrieval using the expanded query.

Categories and Subject Descriptors: H.3.3 [Information Search and Retrieval]: Retrieval Models

General Terms: Algorithms, Experimentation

Keywords: query expansion, pseudo feedback, robust query expansion, fusion, query drift

1. INTRODUCTION

Pseudo-feedback-based query expansion methods augment a query with terms from the documents most highly ranked by an initial search [4]. While the state-of-the-art approaches post effective performance on average, their performance is sometimes quite inferior to that of using only the original query [2, 6, 5]. One of the causes for this *robustness* problem is *query drift* [11]: the change in underlying "intent" between the original query and its expanded form.

Most approaches for query-drift prevention "emphasize" the query terms when constructing the expanded form [12, 13, 1]. In contrast, we demonstrate the merits in "rewarding" documents that are retrieved in response to the expanded form and that are "faithful" to the original query. Specifically, inspired by work on combining multiple query representations [3] we *fuse* the lists retrieved in response to the original query and to its expanded form.

2. RETRIEVAL FRAMEWORK

We use q, d, and $Score_{\text{init}}(d|q)$ to denote a query, a document, and a score assigned to d in response to q by *some* initial search, respectively; $\mathcal{D}_{\text{init}}$ denotes the list of documents most highly ranked according to $Score_{\text{init}}(d|q)$. We assume that some pseudo-feedback-based query expansion approach uses information from *some* documents in $\mathcal{D}_{\text{init}}$ for ranking the entire corpus and that $PF(\mathcal{D}_{\text{init}})$ is the resultant list of highest ranked documents; $Score_{\text{pf}}(d|q)$ denotes the score assigned to d by the pseudo-feedback-based retrieval[1].

2.1 Algorithms

The following retrieval methods essentially operate on $\mathcal{D}_{\text{init}} \cup PF(\mathcal{D}_{\text{init}})$.

The **combMNZ** method [7] rewards documents that are ranked high in *both* $\mathcal{D}_{\text{init}}$ and $PF(\mathcal{D}_{\text{init}})$:[2]

$$Score_{combMNZ}(d|q) \stackrel{def}{=} (\delta[d \in \mathcal{D}_{\text{init}}] + \delta[d \in PF(\mathcal{D}_{\text{init}})]) \cdot \left(\frac{\delta[d \in \mathcal{D}_{\text{init}}]\, Score_{\text{init}}(d|q)}{\sum_{d' \in \mathcal{D}_{\text{init}}} Score_{\text{init}}(d'|q)} + \frac{\delta[d \in PF(\mathcal{D}_{\text{init}})]\, Score_{\text{pf}}(d|q)}{\sum_{d' \in PF(\mathcal{D}_{\text{init}})} Score_{\text{pf}}(d'|q)}\right).$$

Note that a document that belongs to only one of the two lists ($\mathcal{D}_{\text{init}}$ and $PF(\mathcal{D}_{\text{init}})$) can still be among the highest ranked documents.

The **interpolation** algorithm, which was used for preventing query drift in cluster-based retrieval [8], differentially weights the initial score and the pseudo-feedback-based score using an interpolation parameter λ:

$$Score_{interpolation}(d|q) \stackrel{def}{=} \frac{\lambda\delta[d \in \mathcal{D}_{\text{init}}]\, Score_{\text{init}}(d|q)}{\sum_{d' \in \mathcal{D}_{\text{init}}} Score_{\text{init}}(d'|q)} + \frac{(1-\lambda)\delta[d \in PF(\mathcal{D}_{\text{init}})]\, Score_{\text{pf}}(d|q)}{\sum_{d' \in PF(\mathcal{D}_{\text{init}})} Score_{\text{pf}}(d'|q)}.$$

The **re-rank** method, which was also used in work on cluster-based retrieval [8], re-orders the (top) pseudo-feedback-based retrieval results by the initial scores of documents:

$$Score_{re-rank}(d|q) \stackrel{def}{=} \delta[d \in PF(\mathcal{D}_{\text{init}})]\, Score_{\text{init}}(d|q).$$

3. EVALUATION

We use a standard (unigram) language model approach [9] to create the list $\mathcal{D}_{\text{init}}$. Specifically, we set $Score_{\text{init}}(d|q) \stackrel{def}{=} \exp(-CE\left(p_q^{Dir[0]}(\cdot) \,\middle|\middle|\, p_d^{Dir[\mu]}(\cdot)\right))$, where CE is the cross-entropy and $p_x^{Dir[\mu]}(\cdot)$ is a Dirichlet-smoothed language model (μ is the smoothing parameter) induced from x [9, 14, 8].

We use the *relevance model* RM1 [10] for a pseudo-feedback-based query expansion approach. We construct RM1 from the n documents in $\mathcal{D}_{\text{init}}$ with the highest $Score_{\text{init}}(d|q)$; we use Jelinek-Mercer smoothing with parameter α for the

[1] $Score_{\text{init}}(d|q)$ and $Score_{\text{pf}}(d|q)$ are assumed to be non negative, as is the case in our implementation.
[2] For statement s, $\delta[s] = 1$ if s is true and 0 otherwise.

SIGIR'08, July 20–24, 2008, Singapore.
ACM 978-1-60558-164-4/08/07.

corpus	queries	disks
TREC1-3	51-200	1-3
ROBUST	301-450	
	601-700	4,5
WSJ	151-200	1,2
SJMN	51-150	3
AP	51-150	1-3

	TREC1-3		ROBUST		WSJ		SJMN		AP	
	MAP	$< Init$	MAP	$< Init$	MAP	$< Init$	MAP	$< Init$	MAP	$< Init$
Init. Rank.	14.9	-	25.0	-	27.8	-	18.9	-	22.2	-
RM1	19.2^{i}	38.7	27.5^{i}	45.4	33.2^{i}	34.0	24.1^{i}	37.0	28.5^{i}	38.4
RM3	$\mathbf{20.0}^{i}_{r}$	28.0	$\mathbf{29.9}^{i}_{r}$	33.7	$\mathbf{34.7}^{i}_{r}$	28.0	$\mathbf{24.6}^{i}_{r}$	29.0	$\mathbf{29.1}^{i}$	28.3
combMNZ	18.2^{i}	**24.0**	28.0^{i}	**28.5**	31.1^{i}	**14.0**	21.6^{i}	20.0	26.9^{i}	21.2
interpolation	19.5^{i}	31.3	29.3^{i}_{r}	34.9	34.0^{i}	26.0	23.6^{i}	27.0	28.6^{i}	31.3
re-rank	17.5^{i}_{r}	27.3	26.3^{i}	30.9	29.8^{i}_{r}	22.0	20.4^{i}_{r}	**16.0**	25.9^{i}_{r}	**20.2**

Figure 1: Performance numbers of the initial ranking that is based on using only the original query, the relevance models RM1 and RM3, and the fusion-based methods. Boldface: best result per column; "i" and "r" indicate statistically significant MAP differences with the initial ranking and RM1, respectively.

construction [14]. We use only the β terms to which RM1 assigns the highest probability, and denote the resultant (normalized) distribution by $\tilde{p}_{RM1}(\cdot; n, \alpha, \beta)$ [1]. Then, we set $Score_{\text{pf}}(d|q) = \exp(-CE\left(p_d^{Dir[\mu]}(\cdot) \;\middle|\middle|\; \tilde{p}_{RM1}(\cdot; n, \alpha, \beta)\right))$.

We use RM3 [1] as a reference comparison for our methods. RM3 performs *query-anchoring* at the language model level by interpolating (with parameter λ) $\tilde{p}_{RM1}(\cdot; n, \alpha, \beta)$ with a maximum likelihood estimate of the query terms.

3.1 Experimental setup

We used the TREC corpora from Figure 1 for experiments. (Topics' titles serve as queries.) We applied Porter stemming via the Lemur toolkit (www.lemurproject.org), and removed INQUERY stopwords.

We set $\mathcal{D}_{\text{init}}$ to the 1000 documents with the highest *initial ranking* score $Score_{\text{init}}(d|q)$. To create a set $PF(\mathcal{D}_{\text{init}})$ of 1000 documents, we select the values of RM1's free parameters from the following sets so as to optimize MAP@1000 (henceforth "MAP") performance: $n \in \{25, 50, 75, 100, 500, 1000\}$, $\alpha \in \{0, 0.1, 0.2, 0.3\}$, and $\beta \in \{25, 50, 75, 100, 250, 500, 1000\}$. λ, which controls query-anchoring in the interpolation and RM3 algorithms, is chosen from $\{0.1, \ldots, 0.9\}$ to optimize MAP; μ is set to 1000 [14].

We determine statistically significant MAP differences using Wilcoxon's two-tailed test at a confidence level of 95%. We also present for each method the percentage of queries (denoted by "$< Init$") for which the (M)AP performance is *worse* than that of the initial ranking. Lower values of "$< Init$" correspond to improved robustness.

3.2 Experimental results

We see in Figure 1 that all fusion-based methods yield MAP performance that is better to a statistically significant degree than that of the initial ranking that utilizes only the original query. The interpolation algorithm is the best MAP performing fusion-based method, but it incorporates a free parameter while combMNZ and re-rank do not.

Figure 1 also shows that all fusion-based methods are more robust than RM1. (Refer to the "$< Init$" measure.) Furthermore, combMNZ and interpolation post MAP performance that is never worse to a statisticaly significant degree than that of RM1. We also observe that combMNZ and re-rank, which use fusion of retrieved results for query-anchoring, are more robust than RM3 that performs language-model-based query-anchoring; RM3, however, posts the best MAP performance in Figure 1.

4. CONCLUSION

Fusing the lists retrieved in response to a query and to its expanded form can significantly outperform retrieval based on the query alone. The resultant performance is also consistently more robust than that of using the expanded query form, and (for two of the tested fusion-based methods) is more robust than that of performing query-anchoring when creating an expanded form.

Acknowledgments We thank the reviewers for their comments. This paper is based upon work supported in part by the Jewish Communities of Germany Research Fund and by a gift from Google. Any opinions, findings and conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect those of the sponsoring institutions.

5. REFERENCES

[1] N. Abdul-Jaleel, J. Allan, W. B. Croft, F. Diaz, L. Larkey, X. Li, M. D. Smucker, and C. Wade. UMASS at TREC 2004 — novelty and hard. In *Proceedings of TREC-13*, 2004.

[2] G. Amati, C. Carpineto, and G. Romano. Query difficulty, robustness, and selective application of query expansion. In *Proceedings of ECIR*, pages 127–137, 2004.

[3] N. J. Belkin, C. Cool, W. B. Croft, and J. P. Callan. The effect of multiple query representations on information retrieval system performance. In *Proceedings of SIGIR*, pages 339–346, 1993.

[4] C. Buckley, G. Salton, J. Allan, and A. Singhal. Automatic query expansion using SMART: TREC3. In *Proceedings of TREC-3*, pages 69–80, 1994.

[5] K. Collins-Thompson and J. Callan. Estimation and use of uncertainty in pseudo-relevance feedback. In *Proceedings of SIGIR*, pages 303–310, 2007.

[6] S. Cronen-Townsend, Y. Zhou, and W. B. Croft. A language modeling framework for selective query expansion. Technical Report IR-338, Center for Intelligent Information Retrieval, University of Massachusetts, 2004.

[7] E. A. Fox and J. A. Shaw. Combination of multiple searches. In *Proceedings of TREC-2*, 1994.

[8] O. Kurland and L. Lee. Corpus structure, language models, and ad hoc information retrieval. In *Proceedings of SIGIR*, pages 194–201, 2004.

[9] J. D. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of SIGIR*, pages 111–119, 2001.

[10] V. Lavrenko and W. B. Croft. Relevance-based language models. In *Proceedings of SIGIR*, pages 120–127, 2001.

[11] M. Mitra, A. Singhal, and C. Buckley. Improving automatic query expansion. In *Proceedings of SIGIR*, pages 206–214, 1998.

[12] J. J. Rocchio. Relevance feedback in information retrieval. In G. Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*, pages 313–323. Prentice Hall, 1971.

[13] C. Zhai and J. D. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of CIKM*, pages 403–410, 2001.

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