Revisiting Iterative Relevance Feedback for Document and Passage Retrieval

Keping Bi
University of Massachusetts
Amherst, MA
kbi@cs.umass.edu

Qingyao Ai
University of Massachusetts
Amherst, MA
aiqy@cs.umass.edu

Bruce Croft
University of Massachusetts
Amherst, MA
croft@cs.umass.edu

ABSTRACT

As more and more search traffic comes from mobile phones, intelligent assistants, and smart-home devices, new challenges (e.g., limited presentation space) and opportunities come up in information retrieval. Previously, an effective technique, relevance feedback (RF), has rarely been used in real search scenarios due to the overhead of collecting users' relevance judgments. However, since users tend to interact more with the search results shown on the new interfaces, it becomes feasible to obtain users' assessments on a few results during each interaction. This makes iterative relevance feedback (IRF) techniques look promising today. IRF can deal with a simplified scenario of conversational search, where the system asks users to provide relevance feedback on results shown in the current iteration and shows more relevant results in the next interaction. IRF has not been studied systematically in the new search scenarios and its effectiveness is mostly unknown. In this paper, we re-visit IRF and extend it with RF models proposed in recent years. We conduct extensive experiments to analyze and compare IRF with the standard top-k RF framework on document and passage retrieval. Experimental results show that IRF is at least as effective as the standard top-k RF framework for documents and much more effective for passages. This indicates that IRF for passage retrieval has huge potential and is a promising direction for conversational search based on relevance feedback.

KEYWORDS

Iterative Relevance Feedback; Document Retrieval; Passage Retrieval

ACM Reference Format:
Keping Bi, Qingyao Ai, and Bruce Croft. 2019. Revisiting Iterative Relevance Feedback for Document and Passage Retrieval. In *Proceedings of WCIS'19*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

Recently, the interface of modern search engines has experienced significant changes. More than 50% of search traffic comes from mobile phones in 2018 [1], and the number of people who use intelligent assistants (e.g., Siri) and smart-home devices (e.g., Echo) for search is also increasing today. On the one hand, new search environments introduce new challenges to search engines. For example, the precision of top-1 results could significantly affect user experience because assistants or smart-home devices usually present only one result at a time. On the other hand, the modern search scenarios provide new opportunities for the study of interactive search. People tend to interact more with phones and smart-home devices, so deploying relevance feedback (RF) techniques to real search systems becomes feasible and promising.

Relevance feedback has been shown to be effective through extensive studies in the IR community [3, 9, 13, 15, 16, 21]. The idea of RF is to use the explicit relevance judgments provided by users to refine the query model and further retrieve more relevant results. Most existing studies focus on developing an effective RF model that improves the retrieval system in a single iteration, where users assess the relevance of top 10 or more documents in the initial ranking list [16]. Due to significant manual efforts required for relevance judgments, these RF models have been seldom used in real search scenarios.

In new search environments, relevance feedback could be potentially collected through the interactions between users and the system. Figure 1 shows an example conversation between the assistant and a user where the quality of a search result can be obtained during the interaction. Since the display space or bandwidth is severely limited, it is more natural to do re-ranking iteratively after collecting user feedback on a small number of results during the search interactions rather than gathering a lot of feedback and do a one-shot retrieval refinement. We refer to the former as iterative relevance feedback (IRF) and the latter as the standard top-k RF framework.

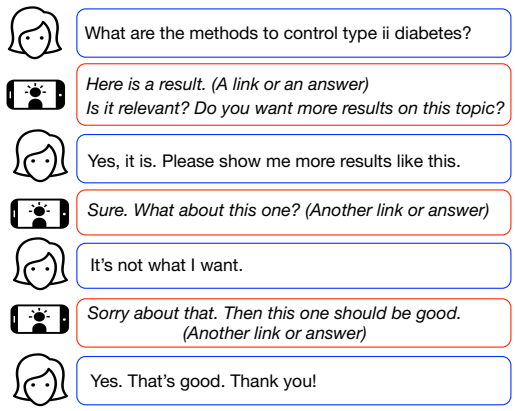

Figure 1: An example conversation on mobile devices where relevance feedback can be obtained and iterative search is preferred.

[1] http://gs.statcounter.com/platform-market-share/desktop-mobile-tablet

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
WCIS'19, July 2019, Paris, France
© 2019 Copyright held by the owner/author(s).
ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

After IRF was proposed and investigated based on Rocchio in the 1990s, many new RF techniques [3, 9, 21] have appeared. However, as far as we know, there is no systematical study on IRF techniques and the effectiveness of IRF with new RF methods in modern search scenarios remains mostly unknown. In this paper, we conduct a systematic study of IRF with more recent models and under different scenarios. Specifically, we focus on two research questions: (1) Given a fixed budget (relevance judgments), does IRF perform better than the standard top-k RF framework for recent RF methods? (2) Does IRF perform equally well in retrieval tasks with different granularities? To answer these questions, we extend several representative RF methods to iterative versions and conduct extensive experiments on two search tasks, document and passage retrieval. The first task aims to simulate the cases where users conduct traditional ad-hoc retrieval (coarse granularity) with limited display space (e.g., phone screen), while the second task focuses on the scenarios where a search engine directly returns the answer or a relevant passage of a query (e.g., search on intelligent assistants or smart-home devices, fine granularity). Experimental results show that IRF works better or at least similar to the standard top-k RF framework on document retrieval and much more effective on passage collections.

2 RELATED WORK

Relevance Feedback. In general, there are three types of relevance feedback (RF) methods for ad-hoc retrieval, which are based on the vector space model (VSM) [17], the probabilistic model [11] and the language model (LM) for information retrieval (IR) [12]. Rocchio [15] is an RF model based on VSM, which refines the vector of a user query by bringing it closer to the center of relevant documents and further from the center of non-relevant documents. In the probabilistic RF method, expansion terms are scored according to the probability of their occurrence in relevant documents compared to non-relevant documents. More recently, feedback techniques have been investigated extensively based on LM, among which, the relevance model [9] and the mixture model [21] are two well-known examples that empirically perform well [10]. Later, the Distillation model [3] adds a query specific non-relevant language models to the mixture model. In addition, there have also been systematical studies on various pseudo RF methods in LM and VSM [6, 10], but no such study on IRF.

Iterative Relevance Feedback. IRF was first proposed by Aalsberg et al. [1] based on Rocchio. In their work, users are asked to judge a single result shown in each interaction, then the query model can be refined iteratively with feedback. This approach showed better performance than standard batch feedback. Later, Allan et al. [2] showed the effectiveness of incremental RF also based on Rocchio for information filtering. Different from their work, we revisit IRF with recently proposed models on retrieval tasks of different granularities.

Some recent TREC tracks [5, 19] also made use of IRF, but their objectives are different and require a large amount of user feedback. The Total Recall track [19] aims to promote all of the relevant documents before non-relevant ones with a huge number of relevance judgments. The target of the Dynamic Domain track [5] is to identify documents satisfying all the aspects of the users' information need with passage-level feedback. In contrast, we investigate IRF with a fixed small amount of feedback and perform a systematical study of IRF for both document and passage retrieval.

3 ITERATIVE RF MODELS

In contrast to top-k RF, in iterative RF, on the one hand, better results may be identified within fewer iterations due to earlier re-ranking, which will reduce the cost of user assessment during search interactions. On the other hand, there are fewer results available with feedback, especially in the first several iterations. RF models require sufficient text to estimate the probabilities or weights of expansion terms that represent the relevance topic model accurately. Little text may be insufficient to distill the non-relevant topics contained in the relevant results and cause topic drift. This problem could be more severe for passages since they are shorter than documents.

To study top-k RF and IRF systematically, we first reformulate some representative RF models based on the language model (LM), the vector space model (VSM) and the probabilistic framework as iterative models. Specifically, we use the relevance model (RM3) [9] and the Distillation model [3] for LM [12]; Rocchio [15] for VSM [17]; and a conventional method with adjusted deviation (Prob) for the probabilistic framework [16]. To generate the initial ranking, we use Query Likelihood (QL) for LM, BM25 for VSM and Prob.

To keep the query model from diverging to non-relevant topics, we maintain two pools for relevant and non-relevant results. Let $RP^{(i)}$ and $NRP^{(i)}$ be the set of all the judgments collected until the ith iteration. Then, in the ith iteration, new judged relevant results $R^{(i)}$ and non-relevant results $NR^{(i)}$ are added to $RP^{(i)}$ and $NRP^{(i)}$, i.e.,

$$RP^{(i)} = RP^{(i-1)} \cup R^{(i)},\ NRP^{(i)} = NRP^{(i-1)} \cup NR^{(i)}$$

where $i > 0, RP^{(0)} = \emptyset, NRP^{(0)} = \emptyset$. We also tried to incrementally estimate the query model in the ith iteration, i.e. $Q^{(i)}$, with $Q^{(i-1)}$, $R^{(i-1)}$ and $NR^{(i-1)}$. This method, however, suffers from topic drift severely and performs much worse than using the original query $Q^{(0)}$ and $RP^{(i-1)}$ and $NRP^{(i-1)}$.

Iterative Relevance Model. RM3 [9] is a well-known pseudo RF method that has also been used for RF. Let $c(w, x)$ be the count of term w in a piece of text x, and $p_x^{MLE}(w) = \frac{c(w,x)}{\sum_{w' \in x} c(w',x)}$ be the maximum likelihood estimate (MLE) of w with respect to x. The relevance model in the ith iteration ($i > 1$)can be estimated with the true RF version of RM3 [8, p. 69] according to

$$p^{(i)}_{rel_{rm3}}(w) = \frac{1}{|RP^{(i-1)}|} \sum_{x \in RP^{(i-1)}} p_x^{MLE}(w) \tag{1}$$

Then, the updated query language model in the ith iteration is the linear combination of the original query language model $p^{MLE}_{Q^{(0)}}(\cdot)$ and $p^{(i)}_{rel_{rm3}}(\cdot)$. Finally, the documents are ranked with the KL divergence between the language models of the query and the documents.

Iterative Distillation Model. Distillation [3] is one of the most recent RF methods, which extends the mixture model [21] by incorporating a query specific non-relevant topic model. It assumes that terms in relevant documents are generated from a mixture of a

Table 1: Statistics of experimental datasets.

Dataset	#Docs	DocLen	Vocab	#Query	#Qrels
Robust	0.5M	504	0.6M	250	17,412
Gov2	25M	893	35M	150	26,917
WebAP	379k	45	59k	80	3843

relevance topic model $p_{rel_{distill}}(\cdot)$, a query specific non-relevance topic model $p_{NR}^{MLE}(\cdot)$, and a background corpus language model $p_C^{MLE}(\cdot)$. For the ith iteration ($i > 1$), $p_{rel_{distill}}^{(i)}(\cdot)$ is estimated with the EM algorithm to maximize the log likelihood of words in $RP^{(i-1)}$, i.e.,

$$\sum_{x \in RP^{(i-1)}} \sum_{w} c(w,x) \log\Big((1-\lambda_1-\lambda_2) p_{rel_{distill}}^{(i)}(w) + \lambda_1 p_{NRP^{(i-1)}}^{MLE}(w) + \lambda_2 p_C^{MLE}(w)\Big) \quad (2)$$

where λ_1 and λ_2 are hyper-parameters. Note that if λ_1 is set to 0, Distillation is exactly the same as the mixture model [21]. Similar to RM3, $p_{rel_{distill}}^{(i)}(\cdot)$ is linear combined with $p_{Q^{(0)}}^{MLE}(\cdot)$ to calculate the new query model for the ith iteration, which then acts as a basis to score results according to KL divergence.

Iterative Rocchio Model. In VSM, queries and documents are represented with vectors in high-dimensional term space. The weight of each dimension can be calculated in many ways and a similarity measure is used to score documents. In this work, we use the BM25 [14] weight for terms in a document or passage vector $\vec{x}$ and dot product as the similarity measure. The term weight in the vector of the initial query $Q^{(0)}$ is set to be the term count in $Q^{(0)}$, i.e., $c(w, Q)$. Then, the query vector in the ith iteration ($i > 0$) is computed as

$$\vec{Q^{(i)}} = \vec{Q^{(0)}} + \beta \frac{1}{|RP^{(i-1)}|} \sum_{x \in RP^{(i-1)}} \vec{x} + \gamma \frac{1}{|NRP^{(i-1)}|} \sum_{x \in NRP^{(i-1)}} \vec{x} \quad (3)$$

where β and γ are the coefficients to balance the influence of postive and negative feedback. If $RP^{(i-1)}$ or $NRP^{(i-1)}$ is empty, the corresponding part is omitted. The relevance score of a document or an answer passage x with respect to a query is computed with the dot product between $\vec{Q^{(i)}}$ and $\vec{x}$.

Iterative Probabilistic Model. In the probabilistic framework [13], the feedback model at ith iteration is estimated by

$$\begin{aligned} p_{prob}^{(i)}(w) &= \log\Big(p_w(1-u_w)/u_w(1-p_w)\Big) \\ p_w &= P(w|rel) = \frac{df_{RP^{(i-1)}}(w) + df_C(w)/|C|}{|RP^{(i-1)}| + 1} \\ u_w &= P(w|nonrel) = \frac{df_C(w) - df_{RP^{(i-1)}}(w) + df_C(w)/|C|}{|C| - |RP^{(i-1)}| + 1} \end{aligned} \quad (4)$$

where $df_S(w)$ is the document frequency of w in the set S (corpus C and $RP^{(i-1)}$ in this case); The term weight of the original query is computed with

$$p_{prob,Q^{(0)}}(w) = \log\Big((|C| - df_C(w))/df_C(w)\Big) \quad (5)$$

The query model at ith iteration is the linear combination between $p_{prob,Q^{(0)}}(\cdot)$ and $p_{prob}^{(i)}(\cdot)$. Again, dot product is used to score documents or passages.

4 IRF EXPERIMENTS

4.1 Experimental Setup

We used standard TREC collections, Robust, Gov2, for document retrieval and WebAP [7, 20] for passage retrieval. Statistics of the datasets are summarized in Table 1. All the methods were implemented based on the Galago toolkit [2]. Stopwords were removed and words were stemmed with Krovetz Stemmer. To compare IRF with typical top-k feedback in a fair manner, we fixed the total number of judged results as 10 and experimented on 1, 2, 5, and 10 iterations, where 10, 5, 2, 1 results were judged during each iteration, respectively. Then, $10D \times 1I$ (10Doc-1Iter) is exactly the top-k feedback. Considering the limitation of presenting results in a real interactive search scenario, we pay more attention to the settings of one or two results per iteration. Users' judgments were simulated by true labels of results.

All the parameters were set using 5-fold cross-validation with grid search. We tuned μ of QL in $\{30, 50, 300, 500, 1000, 1500\}$ and k of BM25 from $\{1.2, 1.4, \cdots, 2\}$. b is set as 0.75. We scanned λ_1, λ_2 in Equation 2 and the interpolation coefficient for the feedback model from $\{0, 0.2, 0.4, \cdots, 1.0\}$, the number of expansion terms m from $\{10, 20, \cdots, 50\}$, and β, γ in equation 3 from $\{0, 0.5, 1, \cdots, 3.0\}$.

Similar to [1], we use freezing ranking [4] to evaluate the performance of IRF. The result lists are formed according to the order they are shown to users during the interactions. Previously shown results are removed in the following retrieval. Results retrieved in the last iteration are appended to the final rank list. Then we use *MAP* at cutoff 1000 and *NDCG*@20 to measure the performance of results overall and on the top. As suggested by Smucker et al. [18], Fisher randomization test with threshold 0.05 is used to calculate statistical significance.

4.2 Results and Discussion

In this section, we discuss and compare the performance of IRF and standard top-k RF feedback in retrieval tasks with different granularities. Table 2 shows the performance of the initial rank lists (QL for RM3 and Distillation, BM25 for Rocchio and Prob), standard top-10 RF (10×1) and the IRF experimental results (5×2, 2×5, 1×10). In general, IRF is effective on both document and passage collections in most cases.

For document retrieval, IRF improves the performance compared with the top-k framework under many iteration settings, but there is no clear correlation between the performance with the number of iterations. This indicates that increasing iteration numbers with a small amount of feedback in each iteration does not always improve the performance. Because documents usually span multiple topics, reducing the number of feedback documents in each iteration makes the ranking system more vulnerable to drift to the non-relevant topics contained in the judged relevant documents. IRF needs enough relevant documents to estimate a robust query model for users' true information need in order to keep the topic from drifting.

The topic drift problem is more severe in Gov2 than in Robust. IRF improves *MAP* significantly in many cases on Robust, but has similar or worse *MAP* on Gov2. The reason could be that Robust is

[2] http://www.lemurproject.org/galago.php

Table 2: Performance of iterative feedback on document and answer passage collections. * and + denote significant improvements over the initial ranked list (Initial) and the standard top-10 feedback model (10×1). The initial ranking model is QL for RM3, Distillation, and BM25 for Rocchio and Prob. Best MAP and $NDCG$ of each method are marked in bold.

Dataset		Method (Doc×Iter)	MAP of freezing rank lists					$NDCG$@20 of freezing rank lists				
			Initial	(10×1)	(5×2)	(2×5)	(1×10)	Initial	(10×1)	(5×2)	(2×5)	(1×10)
Document Retrieval	Robust	RM3	0.253	0.316*	0.321*+	0.321*+	**0.324***+	0.416	0.461*	0.474*+	**0.478***+	**0.478***+
		Distillation	0.253	0.311*	0.321*+	0.322*+	**0.327***+	0.416	0.461*	0.474*+	0.480*+	**0.486***+
		Rocchio	0.255	0.316*	**0.325***+	0.315*	0.316*	0.418	0.463*	**0.476***+	0.462*	0.467*
		Prob	0.255	0.287*	0.287*	**0.288***	0.287*	0.418	0.451*	0.450*	**0.456***+	0.455*
	Gov2	RM3	0.294	**0.349***	0.343*	0.338*	0.337*	0.405	0.451*	**0.464***+	0.454*	0.458*+
		Distillation	0.294	**0.339***	0.337*	0.336*	**0.339***	0.405	0.443*	0.452*+	0.452*+	**0.464***+
		Rocchio	0.295	0.316*	**0.327***+	0.323*+	0.326*+	0.416	0.447*	**0.456***+	0.453*+	0.450*
		Prob	0.295	**0.317***	0.316*	0.314*	0.315*	0.416	0.442*	**0.454***+	0.450*+	0.450*+
Passage Retrieval	WebAP	RM3	0.093	0.115*	0.121*+	**0.132***+	0.130*+	0.143	0.166*	0.174*+	0.186*+	**0.189***+
		Distillation	0.093	0.118*	0.115*	**0.134***+	0.132*+	0.143	0.166*	0.177*+	0.185*+	**0.187***+
		Rocchio	0.101	0.120*	0.134*+	0.138*+	**0.139***+	0.150	0.167*	0.179*+	0.183*+	**0.185***+
		Prob	0.101	0.127*	0.130*	0.134*	**0.138***	0.150	0.170*	0.177*+	0.183*+	**0.192***+

a homogeneous dataset of high-quality news articles and shorter average document length, while Gov2 is a heterogeneous collection of noisy web pages and longer average document length. So more non-relevant information may appear in the judged relevant documents in Gov2 and topic drift is more likely to happen.

Besides, IRF tends to have better performance on top results compared with the overall rank list on document collections. In more cases, $NDCG$@20 is improved by iterative models compared with MAP, especially on Gov2. This indicates that top-ranked results are less suffered from the topic drift problem than results with lower ranking scores.

In contrast to document retrieval, the benefits of IRF for passage retrieval is much more compelling. In Table 2, the performance of IRF on WebAP is positively correlated with the number of iterations. Almost all methods achieve their best results with 10 iterations. Since answer passages are much shorter than documents, they are usually focused on a single topic and less likely to suffer from topic drift. As a result, the whole retrieval system can obtain more improvements when re-ranking is done in an earlier stage, even when we have a limited number of feedback passages. This indicates that IRF techniques could have significant potential for answer passage retrieval.

5 CONCLUSION AND FUTURE WORK

We reformulate feedback models in the three main feedback frameworks as iterative models and investigated the performance of these IRF models on document and passage retrieval. Results show that IRF is at least as effective as standard top-k feedback for retrieving documents and is more powerful in finding answers. For future work, we consider incorporating semantic information to complement word-based IRF models for passage retrieval. We also intend to study how to identify the first relevant answer in fewer iterations based on negative feedback.

6 ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF IIS-1715095. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

[1] IJsbrand Jan Aalbersberg. 1992. Incremental relevance feedback. In *Proceedings of SIGIR'92*. ACM, 11–22.
[2] James Allan. 1996. Incremental relevance feedback for information filtering. In *SIGIR'96*. ACM, 270–278.
[3] Elinor Brondwine, Anna Shtok, and Oren Kurland. 2016. Utilizing focused relevance feedback. In *SIGIR'16*. ACM, 1061–1064.
[4] C Cirillo, Y Chang, and J Razon. 1969. Evaluation of feedback retrieval using modified freezing, residual collection, and test and control groups. *Scientific Report No. ISR-16 to the National Science Foundation* (1969).
[5] Maura R Grossman, Gordon V Cormack, and Adam Roegiest. 2016. TREC 2016 Total Recall Track Overview.. In *TREC'16*.
[6] Kai Hui, Ben He, Tiejian Luo, and Bin Wang. 2011. A comparative study of pseudo relevance feedback for ad-hoc retrieval. In *ICTIR'11*. Springer, 318–322.
[7] Mostafa Keikha, Jae Hyun Park, W Bruce Croft, and Mark Sanderson. 2014. Retrieving passages and finding answers. In *Proceedings of the 2014 Australasian Document Computing Symposium*. ACM, 81.
[8] Victor Lavrenko. 2004. *A Generative Theory of Relevance*. Ph.D. Dissertation. AAI3152722.
[9] Victor Lavrenko and W Bruce Croft. 2017. Relevance-based language models. In *ACM SIGIR Forum*, Vol. 51. ACM, 260–267.
[10] Yuanhua Lv and ChengXiang Zhai. 2009. A comparative study of methods for estimating query language models with pseudo feedback. In *CIKM'09*. 1895–1898.
[11] Melvin Earl Maron and John L Kuhns. 1960. On relevance, probabilistic indexing and information retrieval. *Journal of the ACM (JACM)* 7, 3 (1960), 216–244.
[12] Jay M Ponte and W Bruce Croft. 1998. A language modeling approach to information retrieval. In *SIGIR'98*. ACM, 275–281.
[13] Stephen E Robertson and K Sparck Jones. 1976. Relevance weighting of search terms. *JASIST* 27, 3 (1976), 129–146.
[14] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at TREC-3. *Nist Special Publication Sp* 109 (1995), 109.
[15] Joseph John Rocchio. 1971. Relevance feedback in information retrieval. *The Smart retrieval system-experiments in automatic document processing* (1971).
[16] Gerard Salton and Chris Buckley. 1990. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science* 41 (1990), 288–297.
[17] Gerard Salton, Anita Wong, and Chung-Shu Yang. 1975. A vector space model for automatic indexing. *Commun. ACM* 18, 11 (1975), 613–620.
[18] Mark D Smucker, James Allan, and Ben Carterette. 2007. A comparison of statistical significance tests for information retrieval evaluation. In *CIKM'07*. ACM, 623–632.
[19] Grace Hui Yang and Ian Soboroff. 2016. TREC 2016 Dynamic Domain Track Overview.. In *TREC'16*.
[20] Liu Yang, Qingyao Ai, Damiano Spina, Ruey-Cheng Chen, Liang Pang, W Bruce Croft, Jiafeng Guo, and Falk Scholer. 2016. Beyond factoid QA: Effective methods for non-factoid answer sentence retrieval. In *ECIR'16*. Springer, 115–128.
[21] Chengxiang Zhai and John Lafferty. 2001. Model-based feedback in the language modeling approach to information retrieval. In *CIKM'01*. ACM, 403–410.