A Cluster-Based Resampling Method for Pseudo-Relevance Feedback

Kyung Soon Lee
Department of Computer Engineering
Chonbuk National University
Republic of Korea
selfsolee@chonbuk.ac.kr

W. Bruce Croft
croft@cs.umass.edu

James Allan
allan@cs.umass.edu

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

ABSTRACT

Typical pseudo-relevance feedback methods assume the top-retrieved documents are relevant and use these pseudo-relevant documents to expand terms. The initial retrieval set can, however, contain a great deal of noise. In this paper, we present a cluster-based resampling method to select better pseudo-relevant documents based on the relevance model. The main idea is to use document clusters to find dominant documents for the initial retrieval set, and to repeatedly feed the documents to emphasize the core topics of a query. Experimental results on large-scale web TREC collections show significant improvements over the relevance model. For justification of the resampling approach, we examine relevance density of feedback documents. A higher relevance density will result in greater retrieval accuracy, ultimately approaching true relevance feedback. The resampling approach shows higher relevance density than the baseline relevance model on all collections, resulting in better retrieval accuracy in pseudo-relevance feedback. This result indicates that the proposed method is effective for pseudo-relevance feedback.

Categories and Subject Descriptors

H.3.3 [**Information Storage & Retrieval**]: Relevance Feedback

General Terms

Algorithms, Experimentation

Keywords

Information retrieval, pseudo-relevance feedback, a cluster-based resampling, dominant documents, query expansion

1. INTRODUCTION

Most pseudo-relevance feedback methods (e.g., [12,19,7]) assume that a set of top-retrieved documents is relevant and then learn from the pseudo-relevant documents to expand terms or to assign better weights to the original query. This is similar to the process used in relevance feedback, when actual relevant documents are used [23]. But in general, the top-retrieved documents contain noise: when the precision of the top 10 documents (P@10) is 0.5, 5 of them are non-relevant. This is common and even expected in all retrieval models. This noise, however, can result in the query representation "drifting" away from the original query.

This paper describes *a resampling method using clusters* to select better documents for pseudo-relevance feedback. Document clusters for the initial retrieval set can represent aspects of a query on especially large-scale web collections, since the initial retrieval results may involve diverse subtopics for such collections. Since it is difficult to find one optimal cluster, we use several relevant groups for feedback. By permitting overlapped clusters for the top-retrieved documents and repeatedly feeding *dominant documents* that appear in multiple highly-ranked clusters, we expect that an expansion query can be represented to emphasize the core topics of a query.

This is not the first time that clustering has been suggested as an improvement for relevance feedback. In fact, clustering was mentioned in some of the first work related to pseudo-relevance feedback [1]. Previous attempts to use clusters have not improved effectiveness. The work presented here is based on a new approach to using the clusters that produces significantly better results.

Our motivation for using clusters and resampling is as follows: the top-retrieved documents are a query-oriented ordering that does not consider the relationship between documents. We view the pseudo-relevance feedback problem of learning expansion terms closely related to a query to be similar to the classification problem of learning an accurate decision boundary, depending on training examples. We approach this problem by repeatedly selecting dominant documents to expand terms toward dominant documents of the initial retrieval set, as in the boosting method for a weak learner that repeatedly selects hard examples to change the decision boundary toward hard examples. The hypothesis behind using overlapped document clusters is that a good representative document for a query may have several nearest neighbors with high similarities, participating in several different clusters. Since it plays a central role in forming clusters, this document may be dominant for this topic. Repeatedly sampling dominant documents can emphasize the topics of a query, rather than randomly resampling documents for feedback.

We show that resampling feedback documents based on clusters contributes to higher relevance density for feedback documents on a variety of TREC collections. The results on large-scale web collections such as the TREC WT10g and GOV2 collections show significant improvements over the baseline relevance model.

Conference'04, Month 1–2, 2004, City, State, Country.

The rest of the paper is organized as follows: Section 2 presents related work. Section 3 describes a cluster-based resampling framework. Section 4 shows experimental results on TREC test collections, results analyses and justification of the results. We will conclude in Section 5.

2. RELATED WORK

Our approach is related to previous work on pseudo-relevance feedback, resampling approaches, and the cluster hypothesis in information retrieval.

Relevance feedback (RF) and pseudo-relevance feedback (PRF) have been shown to be effective ways of improving retrieval accuracy by reformulating an original query using relevant or pseudo-relevance documents from the initial retrieval result. New interest in relevance feedback has resulted in the establishment of a relevance feedback track at TREC 2008 [27]. This track will provide a framework for exploring the effects of different factors on relevance feedback, such as initial retrieval, judgment procedure, core reformulation algorithm, and multiple iterations on large scale collection. The motivation of the track shows the current state of research: that relevance feedback is one of the successes of information retrieval over the past 30 years, in that it is applied in a wide variety of settings as both explicit and implicit feedback; however there is surprisingly little new basic research [4]. At the RIA workshop [2], there were comparative experiments on the effects of several factors for pseudo-relevance feedback. The report provides the effects of the number of documents, the number and source of terms used, the initial set of documents, and the effects of swapping documents or terms across systems. In some aspects it is not easy to see real effects, since some factors are mixed up with other effects.

Traditional pseudo-relevance feedback algorithms such as Okapi BM25 [19] and Lavrenko and Croft's relevance model [12] are based on the assumption of relevancy for the top-retrieved documents. Research has been conducted to improve traditional PRF by using passages [29] instead of documents, by using a local context analysis method [28], by using a query-regularized estimation method [26], and by using latent concepts [17]. These methods follow the basic assumption that the top-retrieved documents are relevant to a query.

Recently there has been some work on sampling and resampling techniques for the initial retrieval set. A selective sampling method by Sakai *et al* [22] skips some top-retrieved documents based on a clustering criterion. The cluster is generated not by document similarity but by the same set of query terms. The sampling purpose is to select a more varied and novel set of documents for feedback. Their assumption is that the top-ranked documents may be too similar or redundant. However, their results did not show significant improvements on NTCIR collections. Our approach of repeatedly using dominant documents is based on a different assumption.

A resampling method suggested by Collins-Tompson and Callan [5] uses bootstrap sampling on the top-retrieved documents for the query and variants of the query obtained by leaving a single term out. The assumption behind query variants is that one of the query terms is a noise term. From their experimental analysis, the main gain is from the use of query variants, not document resampling. Their results on robustness and precision at 10 documents (P@10) show improvements, but the performance in terms of mean average precision (MAP) is lower than the baseline relevance model on TREC collections. Our approach primarily focuses on the effects of resampling the top-retrieved documents.

On the other hand, many information retrieval techniques have adopted the cluster hypothesis to improve effectiveness. The cluster hypothesis states that *closely related documents tend to be relevant to the same request* [11]. Re-ranking using clusters [13, 14] based on the vector space model has shown successful results. A cluster-based retrieval model [15] based on language modeling ranks clusters by the likelihood of generating the query. The results show improvements over the query-likelihood retrieval model on TREC collections. A local score regularization method [6] uses a document affinity matrix to adjust initial retrieval scores so that topically related documents receive similar scores. The results on small TREC collections show that regularized scores are significantly better than the initial scores.

There has also been work on term expansion using clustering in the vector space model. At TREC 6, Buckley *et al* [3] used document clustering on SMART though the results of using clusters did not show improvements over the baseline feedback method. At the RIA workshop to investigate the effects criteria for pseudo-relevance feedback [2], there are comparisons to investigate the effects of swapping documents and clusters by document clustering and passage-level clustering. The experimental setup is too complex to see the individual effects of clusters, since an outside source factor is mixed up with the clustering factor [29]: using outside sources for feedback itself affects the performance. Thus the analysis for the comparative experiments is inconclusive.

3. A CLUSTER-BASED RESAMPLING FRAMEWORK FOR FEEDBACK

This section describes the rationale for the method for selective resampling, our resampling procedure, and a justification based on relevance density.

3.1 Selective Resampling Approach

The main issues in pseudo-relevance feedback are how to select relevant documents from the top-retrieved documents, and how to select expansion terms. Here we deal with the problem of selecting better feedback documents.

The problem in traditional pseudo-relevance feedback is obtaining a set of expansion terms from the top-retrieved documents that may have low precision. If a method can select better documents from the given sample, it can almost certainly contribute better expansion terms. For pseudo-relevance feedback, the initial retrieval set can be seen as the sample space of query expansion terms from which we estimate the sampling distribution.

In statistics, resampling (bootstrapping) is a method for estimating the precision of sample statistics by sampling *randomly* with replacement from the original sample, leading to robust estimates. If a method is available for selecting better examples from the original sample space, ***selective sampling*** will perform better than ***random*** sampling. Boosting [24] is a selective resampling method in machine learning. It is an iterative procedure used to *adaptively change the distribution of training examples* so that the weak learners focus on examples that previous weak learners misclassified.

To find some direction to change the distribution, we assume that ***a dominant document for a query*** is one with good representation of the topics of a query—i.e. one with several nearest neighbors with high similarity. In overlapped clusters, a dominant document will appear in multiple highly-ranked clusters. Since a topic can contain several subtopics, the retrieved set can be divided into several subtopic groups. A document that deals with all subtopics will likely be in all subtopic clusters, so we call that document *dominant*. From such a dominant document, expansion terms that retrieve documents related to all subtopics can be selected.

Based on the above assumption, we *selectively resample* documents for feedback using k-nearest neighbors (k-NN) clustering to generate overlapped clusters from the given top-retrieved documents space.

3.2 Resampling Feedback Documents Using Overlapping Clusters

A cluster-based resampling method to get better pseudo-relevant documents is based on the language model [18] and the relevance model [12] frameworks. Relevance models have been shown to be a powerful way to construct a query model from the top-retrieved documents [12, 7]. The essential point of our approach is that a document that appears in multiple highly-ranked clusters will contribute more to the query terms than other documents. The resampling process proceeds as follows:

First, documents are retrieved for a given query by the query-likelihood language model [18] with Dirichlet smoothing [30].

A statistical language model is a probabilistic distribution over all the possible word sequences for generating a piece of text. [19]. In information retrieval, the language model treats documents themselves as models and a query as strings of text generated from these document models. The popular query-likelihood retrieval model estimates document language models using the maximum likelihood estimator. The documents can be ranked by their likelihood of generating or sampling the query from document language models: $P(Q|D)$.

$$P(Q \mid D) = \prod_{i=1}^{m} P(q_i \mid D) \tag{1}$$

where q_i is the i$^{\text{th}}$ query term, m is the number of words in a query Q, and D is a document model.

Dirichlet smoothing is used to estimate non-zero values for terms in the query which are not in a document. It is applied to the query likelihood language model as follows.

$$P(w \mid D) = \frac{|D|}{|D|+\mu} P_{ML}(w \mid D) + \frac{\mu}{|D|+\mu} P_{ML}(w \mid Coll) \tag{2}$$

$$P_{ML}(w \mid D) = \frac{freq(w,D)}{|D|}, \quad P_{ML}(w \mid Coll) = \frac{freq(w,Coll)}{|Coll|} \tag{3}$$

where $P_{ML}(w|D)$ is the maximum likelihood estimate of word w in the document D, $Coll$ is the entire collection, and μ is the smoothing parameter. $|D|$ and $|Coll|$ are the lengths of a document D and collection C, respectively. *freq(w,D)* and *freq(w,Coll)* denote the frequency of a word *w* in *D* and *Coll*, respectively. The smoothing parameter is learned using training topics on each collection in experiments.

Next, clusters are generated by *k*-nearest neighbors (*k*-NN) clustering method [9] for the top-retrieved *N* documents to find dominant documents. (In experiments, *N* is set to 100.) Note that one document can belong to several clusters.

In *k*-NN clustering, each document plays a central role in making its own cluster with its *k* closest neighbors by similarity. We represent a document by *tfidf* weighing and cosine normalization. The cosine similarity is used to calculate similarities among the top-retrieved documents.

Our hypothesis is that a dominant document may have several nearest neighbors with high similarities, participating in several clusters. On the other hand, a non-relevant document ideally makes a singleton cluster with no nearest neighbors with high similarity, though practically it will have neighbors due to noise such as polysemous or general terms. Document clusters can also reflect the association of terms and documents from similarity calculation. In this work, if a document is a member of several clusters and the clusters are highly related to the query, we assume it to be a dominant document. A cluster-based resampling method is repeatedly feeding such dominant documents based on document clusters.

After forming the clusters, we rank them by a cluster-based query-likelihood language model described below [15]. The documents in the top-ranked clusters are used for feedback. Note that clusters are only used for selecting feedback documents.

A cluster can be treated as a large document so that we can use the successful query-likelihood retrieval model. Intuitively, each cluster can be represented by just concatenating documents which belong to the cluster. If *Clu* represents such a cluster, then:

$$P(Q \mid Clu) = \prod_{i=1}^{m} P(q_i \mid Clu) \tag{4}$$

$$P(w \mid Clu) = \frac{|Clu|}{|Clu|+\lambda} P_{ML}(w \mid Clu) + \frac{\lambda}{|Clu|+\lambda} P_{ML}(w \mid Coll) \tag{5}$$

$$P_{ML}(w \mid Clu) = \frac{freq(w,Clu)}{|Clu|}, \quad P_{ML}(w \mid Coll) = \frac{freq(w,Coll)}{|Coll|} \tag{6}$$

where *freq(w,Clu)* is sum of *freq(w, D)* for the document *D* which belongs to the cluster *Clu*.

Finally, expansion terms are selected using the relevance model for each document in the top-ranked clusters. Note that the set of feedback documents chosen from the top-ranked clusters are used to estimate the relevance model with their initial query-likelihoods.

A relevance model is a query expansion approach based on the language modeling framework. The relevance model [12] is a multinomial distribution which estimates the likelihood of words *w* given a query *Q*. In the model, the query words $q_1 \ldots q_m$ and the words *w* in relevant documents are sampled identically and independently from a distribution *R*. Following that work, we estimate the probability of a word in the distribution *R* using

$$\sum_{D \in R} P(D)P(w \mid D)P(Q \mid D) \qquad (7)$$

where *R* is the set of documents that are pseudo-relevant to the query *Q*. We assume that *P(D)* is uniform over the set.

After this estimation, the most likely *e* terms from *P(w|R)* are chosen as an expansion query for an original query. The final expanded query is combined with the original query using linear interpolation, weighted by parameter λ. The combining parameter is learned using training topics on each collection in experiments.

The original relevance model and traditional pseudo-relevance feedback methods use the initial retrieval set to get expansion terms directly after the first step. The problem is that the top-retrieved documents contain non-relevant documents, which add noise to expansion terms. Our effort uses overlapping clusters to find dominant documents for the query. It may still find non-relevant documents, but we will show it finds fewer.

3.3 Justification by Relevance Density

The rationale for the proposed method is that resampling documents using clusters is an effective way to find dominant documents for a query from the initial retrieval set. We measure relevance density to justify our assumption that dominant documents are relevant to the query and redundantly appear over the top-ranked clusters.

The relevance density is defined to be the proportion of the feedback documents that contain relevant documents.

$$Density = \frac{\textit{the number of relevant feedback documents}}{\textit{the number of feedback documents}} \qquad (8)$$

A higher relevance density implies greater retrieval accuracy, ultimately approaching true relevance feedback.

If a resampling method is effective, it will produce higher relevance densities for pseudo-relevant documents than a set of top-retrieved documents. To justify the cluster-based resampling method, we will examine the relevance density of feedback documents through experimental analysis.

4. EXPERIMENTS

To validate the proposed method, we performed experiments on five TREC collections and compared the results with a baseline retrieval model, a baseline feedback model, and an upper-bound model.

4.1 Experimental Set-up

4.1.1 Test Collections

We tested the proposed method on homogeneous and heterogeneous test collections: the ROBUST, AP and WSJ collections are smaller and contain newswire articles, whereas GOV2 and WT10G are larger web collections. For all collections, the topic title field is used as the query. A summary of the test collections is shown in Table 1.

Version 2.3 of the Indri system [25] is used for indexing and retrieval. All collections are stemmed using a Porter stemmer. A standard list of 418 common terms is removed at retrieval time.

Table 1. Training and Test collections

Collection	Description	# of docs	Topics	
			Train	Test
GOV2	2004 crawl of .gov domain	25,205,179	701-750	751-800
WT10g	TREC web collection	1,692,096	451-500	501-550
ROUBST	Robust 2004 collection	528,155	301-450	601-700
AP	Association Press 88-90	242,918	51-150	151-200
WSJ	Wall street Journal 87-92	173,252	51-150	151-200

4.1.2 Training and Evaluation

For each test collection, we divide topics into training topics and test topics, where the training topics are used for parameter estimation and the test topics are used for evaluation.

In order to find the best parameter setting we sweep over values for smoothing parameter to construct the language model (μ ∈ {500, 750, 1000, 1500, 2000, …, 5000}), to construct the relevance model for the number of feedback documents (|*R*| ∈ {5, 10, 25, 50, 75, 100}), the number of expansion terms (*e* ∈ {10, 25, 50, 75, 100}, and the weight of the original query (λ ∈ {0.1, 0.2, …, 0.9}). To train the proposed model, we sweep over the number of feedback clusters (|*C*| ∈ {1, 2, 5, 10, 15, 20}), which corresponds to the number of feedback documents since one cluster can have at most five documents as a member (*k* = 5) in our *k*-nearest neighbors clustering. The threshold for clustering is set to 0.25. Expansion terms are represented using the following Indri query form:

$$\begin{aligned} &\#weight\ \ (\lambda\ \#combine\ \ (q_1 \ldots q_m) \\ &\qquad (1-\lambda)\,\#weight\ \ (p_1\, t_1 \ldots p_e\, t_e)) \end{aligned}$$

where $q_1 \ldots q_m$ are the original query terms, $t_1 \ldots t_e$ are the *e* terms with expansion probabilities $p_1 \ldots p_e$, and λ is a parameter combining the original query and the expanded query.

All comparison methods are optimized on the training set using mean average precision (MAP) defined as,

$$MAP = \frac{1}{|Q|}\sum_{q \in Q} ap(q) \qquad (9)$$

where *ap*(*q*) is average precision for a query in the topic set *Q*.

The best parameters on training for each test collection are used for experimental results with the test topics.

4.1.3 Baselines

We provide two baselines: the language model and the relevance model.

- Language Model (*LM*): The performance of the baseline retrieval model. The relevance model and the resampling method are also based on this framework.
- Relevance Model (*RM*): The performance of the baseline pseudo-relevance feedback model. The expanded query is combined with the original query. The resampling method is

based on the relevance model framework. The difference is the pseudo-relevant documents used.

4.1.4 Upper-bound: True Relevance Feedback

To investigate the performance of the upper-bound of the proposed method, we compare with true relevance feedback.

- True Relevance Feedback (*TrueRF*): The performance using true relevant documents in the top-retrieved 100 documents. This performance presents the upper-bound when using relevance model.

4.1.5 A Cluster-based Reranking Method

To provide the effectiveness of clusters for the initial retrieval set, we also include a cluster-based reranking method.

- Reranking using clusters (*Rerank*): The performance of reranking by combining query likelihoods for documents and clusters based on *k*-NN clusters for the top-retrieved *N* documents. *N* and *k* are set to 1000 and 5, respectively.

$$P'(Q \mid D) = P(Q \mid D) \cdot MAX_{D \in Clu_i} P(Q \mid Clu_i) \qquad (10)$$

Since a document can be a member of several clusters, we choose the maximum query likelihood for the clusters *Clu* which the document *D* belongs to.

4.2 Experimental Results

The results for the comparison methods on five test collections are presented in Table 2.

The *Resampling* method significantly outperforms *LM* on all test collections, whereas *RM* does not significantly outperform *LM* on the WT10g collection. For the GOV2 and WT10g heterogeneous web test collections, the *Resampling* method significantly outperforms *RM*. The relative improvements over *RM* are 6.28% and 19.63% on GOV2 and WT10g, respectively. For the ROBUST newswire collection, the *Resampling* method shows slightly lower performance than *RM*. For the AP and WSJ newswire collections, the *Resampling* method shows small, but not significant improvements over *RM*.

In the precision at 5 (P@5) evaluation metric (not shown in the table), the *Resampling* method shows 14.8%, 24.7%, 3.9%, 20.0%, and 11.9% improvements over *LM*, whereas *RM* shows -7.1%, 7.4%, 1.6%, 18.8% and 7.4% improvement on GOV2, WT10g, ROBUST04, AP and WSJ, respectively.

The *Rerank* method using clusters shows significant improvements over *LM* on all test collections. In fact, the *Rerank* method outperforms *RM* on WT10g collection. The results indicate that document clustering can help find relevant document groups for the initial retrieval set and provide implicit document context to the query.

TrueRF shows significant improvements over all methods on test collections. The results provide upper-bound performance on each collection, when we are able to choose better pseudo-relevant documents, approaching to true relevant documents.

We have also examined the effectiveness as the number of feedback documents and terms varies. As shown in Figure 1, *Resampling* achieves better performance over *RM* for most values. The best parameters selected for feedback on GOV2 are 10 documents and 50 terms for *RM*, 25 documents and 100 terms for *Resampling*. For test topics using the best parameters (μ, *e*, and λ) chosen from training, the *Resampling* method outperforms *RM* regardless of the number of feedback documents.

Table 2. Performance comparisons using mean average precision for the test topics on test collections. The superscripts α, β, γ and δ indicate statistically significant improvements over LM, Rerank, RM and Resampling, respectively. We use the paired *t*-test with significance at $p <$ 0.05.

	LM	Rerank	RM	Resampling	TrueRF
GOV2	0.3258	0.3406^{α}	$0.3581^{\alpha\beta}$	$0.3806^{\alpha\beta\gamma}$	$0.4315^{\alpha\beta\gamma\delta}$
WT10g	0.1861	0.2044^{α}	0.1966	$0.2352^{\alpha\beta\gamma}$	$0.4030^{\alpha\beta\gamma\delta}$
ROBUST	0.2920	0.3206^{α}	$0.3591^{\alpha\beta}$	$0.3515^{\alpha\beta}$	$0.5351^{\alpha\beta\gamma\delta}$
AP	0.2077	0.2361^{α}	$0.2803^{\alpha\beta}$	$0.2906^{\alpha\beta}$	$0.4253^{\alpha\beta\gamma\delta}$
WSJ	0.3258	0.3611^{α}	$0.3967^{\alpha\beta}$	$0.4033^{\alpha\beta}$	$0.5306^{\alpha\beta\gamma\delta}$

Table 3. Performance on fixed feedback documents. The number of feedback documents and terms are both set to 100. The superscripts α and β indicate statistically significant improvements over LM and RM, respectively. We use the paired *t*-test using significance at $p < 0.05$.

	LM	RM	chg%	Resampling	chg%
GOV2	0.3258	0.3519^{α}	8.01	$0.3764^{\alpha\beta}$	15.53
WT10G	0.1861	0.1886	1.34	0.2072^{α}	11.34
ROBUST	0.2920	0.3262^{α}	11.71	$0.3549^{\alpha\beta}$	21.54
AP	0.2077	0.2758^{α}	32.79	0.2853^{α}	37.36
WSJ	0.3258	0.3785^{α}	16.18	$0.4009^{\alpha\beta}$	23.05

4.3 Relevance Density

In this section we aim to develop a deeper understanding of why expansion by the cluster-based resampling method helps.

For justification of a cluster-based resampling approach using overlapping clusters, we have analyzed the relevance density by dominant documents and the performance of feedback without redundant documents.

4.3.1 Relevance Density of Feedback Documents

We can expect that higher relevance density produces higher performance since more relevant documents are used for feedback.

As shown in Figure 2, the resampling method shows higher density compared to the relevance model for all test collections. (The density for AP and WSJ collections is not shown but has the same pattern as the ROBUST collection.)

When the number of feedback documents is set to 100, we can expect that the resampling method outperforms the relevance model since the resampling method uses more relevant documents for feedback.

To verify our expectation for density, we compared performance with the number of feedback documents and terms set to 100. The performance of feedback on fixed documents is

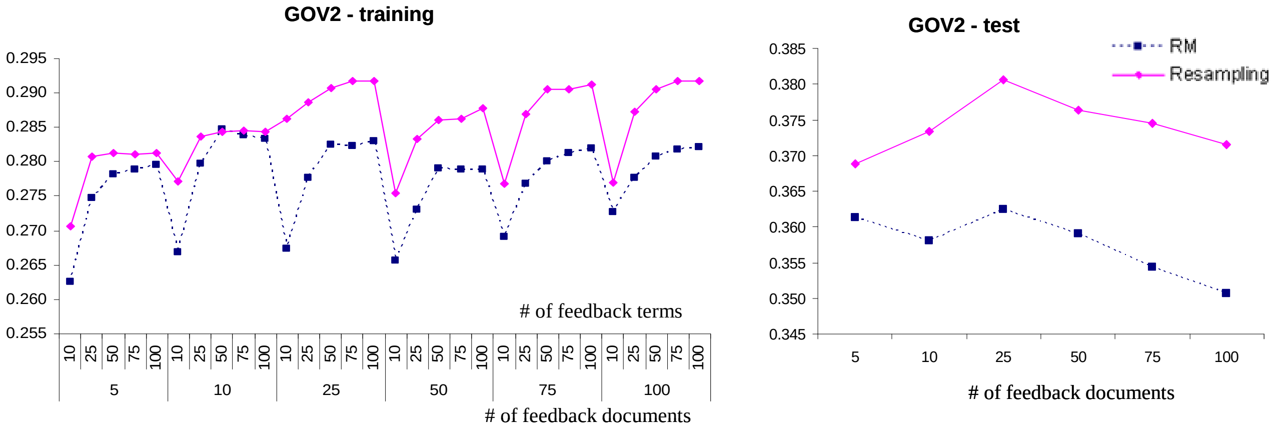

Figure 1. Performances on training and test set for *RM* and *Resampling* according to the number feedback documents and terms on GOV2 collection (in mean average precision).

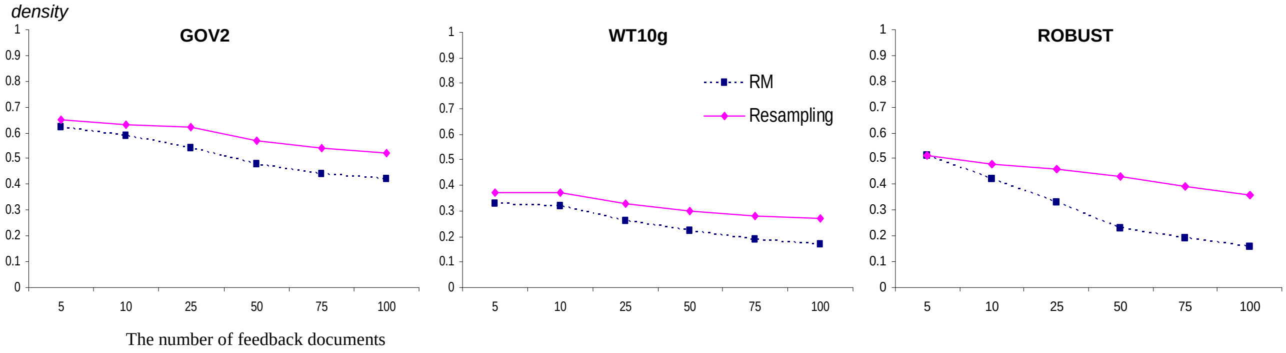

Figure 2. The relevance density for RM and Resampling according to the number of feedback documents.

shown in Table 3. The resampling method outperforms the relevance model for all collections. The results show that the density of relevant documents supports the improvements from the resampling approach which extracts better feedback documents from the top-ranked 100 documents.

From the results of density according to number of feedback documents and effectiveness on all collections, we can conclude that the redundant dominant documents help the density of the relevant documents.

4.3.2 Feedback without redundant documents

To support the observation of relevance density and performance, we have examined performance by removing redundant documents in feedback. That is, a document is not repeated in the feedback even if it occurs in multiple clusters.

We assumed that dominant documents for the initial retrieval set are relevant and redundant documents that play a central role in making overlapping clusters. Table 4 shows the performance of *Sampling without Redundancy*. It outperforms *RM*, but is worse than *Resampling*. The results show that redundant documents give positive effects for feedback.

Table 4. The effect of redundant documents for feedback.

	GOV2	chg%	WT10g	chg%
LM	0.3258	-	0.1861	-
Rerank	0.3406	4.54	0.2044	9.83
RM	0.3581	9.91	0.1966	5.64
Resampling	0.3806	16.82	0.2352	26.38
Sampling without redundancy	0.3745	14.95	0.2193	17.84

We have also examined how redundancy affects the number of relevant documents in the feedback sample. If we look at the top 5, 10, 25, 50, 75, and 100 documents, we find the following. For the *RM* approach, the relevance density was 0.6, 0.5, 0.36, 0.3, and 0.25, respectively. For *Resampling*, however, the densities were almost perfect: 1.0, 1.0, 0.96, 0.98, 0.97, and 0.89, respectively. To illustrate the level of redundancy, consider one query where the top 10 clusters contained 50 documents, 40 of which were relevant: 37 of those relevant documents appeared in other clusters. One relevant document appeared in nine of the top 10 clusters and another was in seven. Some documents that

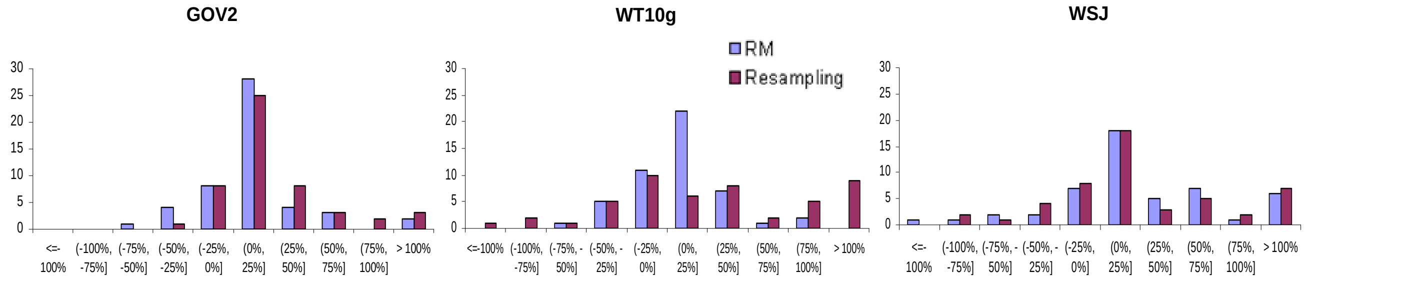

Figure 3. Robustness of the relevance model and the resampling method over the language model for GOV2, WT10g, and WSJ collections.

appear in multiple highly-ranked clusters and their redundancy contribute to query expansion terms.

4.4 Retrieval Robustness

We analyze the robustness of the baseline feedback model and the resampling method over the baseline retrieval model. Here, robustness is defined as the number of queries whose performance is improved or hurt as the result of applying these methods.

Figure 3 presents an analysis of the robustness of the baseline feedback model and the resampling method on GOV2, WT10g and WSJ. The robustness of ROBUST and AP showed the similar pattern with WSJ. For the homogeneous newswire collections such as WSJ, AP and ROBUST, the relevance model and resampling method showed a similar pattern for robustness.

The resampling method shows strong robustness for each test collection. For the GOV2 collection, the resampling method improves 41 queries and hurts 9, whereas the relevance model improves 37 and hurts 13. For the WT10g collection, the resampling method improves 30 and hurts 19, whereas the relevance model improves 32 and hurts 17. Although the relevance model improves the performance of 2 more queries than the resampling method, the improvements obtained by the resampling method are significantly larger. For the ROBUST collection, the resampling method improves 63 and hurts 36, whereas the relevance model improves 64 and hurts 35.

Overall, our resampling method improves the effectiveness for 82%, 61%, 63%, 66% and 70% of the queries for GOV2, WT10g, ROBUST, AP and WSJ, respectively.

5. CONCLUSIONS

Resampling the top-ranked documents using clusters is effective for pseudo-relevance feedback. The improvements obtained were consistent across nearly all collections, and for large web collections, such as GOV2 and WT10g, the approach showed substantial gains. The relative improvements on GOV2 collection are 16.82% and 6.28% over LM and RM, respectively. The improvements on the WT10g collection are 19.63% and 26.38% over LM and RM, respectively. We showed that the relevance density was higher than the baseline feedback model for all test collections as a justification of why expansion by the cluster-based resampling method helps. Experimental results also show that overlapping clusters are helpful for identifying dominant documents for a query.

For future work, we will study how the resampling approach can adopt query variants by considering query characteristics. Additionally, in our experiments we simply represent a cluster by concatenating documents. Using a better representation of a cluster should improve the performance of pseudo-relevance feedback by improving the cluster ranking.

6. ACKNOWLEDGMENTS

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD)" (KRF-2006-611-D00025) and in part by the Center for Intelligent Information Retrieval. 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.

7. REFERENCES

[1] Attar, R. and Fraenkel, A. S. 1977. Local Feedback in Full-Text Retrieval Systems. *Journal of the ACM* 24, 3 (Jul. 1977), pp. 397-417.

[2] Buckley, C. and Harman, D. 2004. Reliable information access final workshop report. http://nrrc.mitre.org/NRRC/publications.htm

[3] Buckley, C., Mitra, M., Walz, J., and Cardie, C. 1998. Using clustering and superconcepts within SMART: TREC 6. In Proc. 6th Text REtrieval Conference (TREC-6).

[4] Buckley, C. and Robertson, S. 2008. Proposal for relevance feedback 2008 track. http://groups.google.com/group/trec-relfeed.

[5] Collins-Thompson, K., and Callan, J. 2007. Estimation and use of uncertainty in pseudo-relevance feedback. In Proc. 30th ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 303-310.

[6] Diaz, F. 2005. Regularizing ad hoc retrieval scores. In Proc. 14th ACM international conference on Information and knowledge management, pp. 672-679.

[7] Diaz, F., and Metzler, D. 2006. Improving the Estimation of Relevance Models Using Large External Corpora, In Proc. 29th ACM SIGIR Conf. on Research and Development on Information Retrieval, pp. 154-161.

[8] Efron, B. 1979. Bootstrap methods: Another look at the jackknife, The Annals of Statistics, 7, pp. 1-26.

[9] Fix, E. and Hodges, L. 1951. Discriminatory analysis: nonparametric discrimination: consistency properties. Technical Report, USAF School of Aviation Medicine, Randolph Field, Texas, Project 21-49-004.

[10] Freund, Y. 1990. Boosting a weak learning algorithm by majority. In Proc. 3rd Annual Workshop on Computational Learning Theory.

[11] Jardine. N. and Rijsbergen, C.J.V. 1971. The use of hierarchic clustering in information retrieval. Information Storage and Retrieval, 7, pp.217-240.

[12] Lavrenko, V. and Croft, W.B. 2001. Relevance-based language models. In Proc. 24th ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 120-127.

[13] Lee, K.S., Park, Y.C., and Choi, K.S. 2001. Re-ranking model based on document clusters. Information Processing and Management, 37, pp. 1-14.

[14] Lee, K.S., Kageura, K., and Choi, K.S. 2004. Implicit ambiguity resolution based on cluster analysis in cross-language information retrieval, Information Processing and Management, 40, pp. 145-159.

[15] Liu, X., and Croft, W.B. 2004. Cluster-based retrieval using language models. In Proc. 27th ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 186-193.

[16] Lynam, T., Buckley, C., Clarke, C., and Cormack, G. 2004. A multi-system analysis of document and term selection for blind feedback. In Proc. 13th ACM international conference on Information and knowledge management (CIKM), pp. 261-269.

[17] Metzler, D. and Croft, W. B. 2007. Latent Concept Expansion Using Markov Random Fields, In Proc. 30th ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 311-318.

[18] Ponte, J.M, and Croft, W.B. 1998. A language modeling approach to information retrieval. In Proc. 21st ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 275–281.

[19] Robertson, S.E., Walker, S., Beaulieu, M., Gatford, M., and Payne, A. 1996. Okapi at TREC-4. In Proc. 4th Text REtrieval Conference (TREC-4).

[20] Rocchio, J.J. 1971. Relevance feedback in information retrieval. The SMART retrieval system, Prentice-Hall, pp. 316-321.

[21] Rosenfeld, R. 2000. Two decades of statistical language modeling: where do we go from here? In Proc. of the *IEEE*, 88(8), pp. 1270-1278.

[22] Sakai, T., Manabe, T. and Koyama, M. 2005. Flexible pseudo-relevance feedback via selective sampling. ACM Transactions on Asian Language Information Processing (TALIP), 4(2), pp. 111-135.

[23] Salton, G., and Buckley, C. 1990. Improving retrieval performance by relevance feedback. Journal of the American Society for Information Science, 41(4), pp. 288-297.

[24] Schapire, R. 1990. Strength of weak learnability. Journal of Machine Learning, 5, pp. 197-227.

[25] Strohman, T., Metzler, D., Turtle, H., and Croft, W.B. 2005. Indri: A language model-based search engine for complex queries. In Proc. International Conference on Intelligence Analysis.

[26] Tao, T., and Zhai, C. 2006. Regularized estimation of mixture models for robust pseudo-relevance feedback. In Proc. 29th ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 162-169.

[27] TREC. 20008. Call for participation. http://trec.nist.gov/call08.html

[28] Xu, J and Croft, W.B. 1996. Query expansion using local and global document analysis. In Proc. 19th ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 4-11.

[29] Yeung, D.L., Clarke, C.L.A., Cormack, G.V., Lynam, T.R., and Terra, E.L. 2004. Task-specific query expansion. In Proc. 12th Text REtrieval Conference (TREC-12), pp. 810-819.

[30] Zhai, C., and Lafferty, J. 2004. A study of smoothing methods for language models applied to information retrieval. ACM Transactions on Information Systems, 22(2), pp.179-214