EXPERIMENTS ON RETRIEVAL OF OPTIMAL CLUSTERS

Xiaoyong Liu
Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts, Amherst, MA 01003
xliu@cs.umass.edu

W. Bruce Croft
Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts, Amherst, MA 01003
croft@cs.umass.edu

ABSTRACT

The most common approach to cluster-based retrieval is to retrieve one or more clusters in their entirety to a query. The system's goal is to assign top ranks to the clusters that give best retrieval performance, out of all clusters. Previous research in this area has suggested that "optimal" clusters exist that, if retrieved, would yield very large improvements in effectiveness relative to document-based retrieval. However, it is precisely if and how the optimal clusters can be identified and used that has long been an interesting and challenging problem. In this paper, we provide a detailed analysis of the characteristics of optimal clusters and propose a new technique that allows the retrieval system to decide whether to use cluster-based or document-based retrieval for a given query. Experiments show that improvements over using either type of retrieval alone can be obtained in a fully automatic manner and without relevance information provided by human.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Retrieval models.*

General Terms: Experimentation

Keywords: Cluster-based Retrieval, Optimal Cluster, Optimal Cluster Retrieval.

1. INTRODUCTION

Cluster-based retrieval has been studied for many years. The most common approach has been based on the notion of *cluster-based retrieval* introduced by Jardine and van Rijsbergen [10]. The task for the retrieval system is to retrieve one or more clusters in their entirety to a query, by matching the query against clusters of documents instead of individual documents and ranking clusters based on their similarity to the query. Jardine and van Rijsbergen introduced the notion of an "optimal" cluster. A cluster is considered optimal if, when retrieved, it would give the maximum possible value for a retrieval criterion out of all clusters. They and others showed that if the optimal clusters could be retrieved, effectiveness would be far better than a document-based search.

Various cluster retrieval and search methods have been proposed [2, 21, 23], and a variety of clustering algorithms have been investigated [23, 8, 12, 18]. Document clustering has been performed either in a static manner over the entire collection, independent of the user's query, or in a query-specific manner in which documents to be clustered are from the retrieval result of a document-based retrieval on the query. A number of studies [9, 10, 18] have examined the quality of optimal clusters and suggested that if the retrieval system were able to find them, retrieval performance can be improved over document-based retrieval. However, no real retrieval strategy has achieved this result. Lower performance was reported in [8, 22, 25]. Based on this, it was speculated that different retrieval strategies might be helpful for different queries. Attempts have been made in automatic selection of retrieval strategies, including the choice between cluster and non-cluster searches [4, 8]. The results showed, however, that it was less successful than consistently using just one strategy.

Recent work on cluster-based retrieval in the language modeling framework [13, 11] has demonstrated that using clusters can help retrieval, but the clusters are used as a form of document smoothing and the system's goal is to rank documents, not clusters. The identification and use of optimal clusters were not addressed.

In this paper we discuss the performance of optimal cluster retrieval on TREC test collections and provide a detailed analysis of why optimal clusters are often not retrieved with state-of-the-art retrieval techniques. It is the first of such an analysis that is reported. We propose a new technique for a system to automatically decide whether cluster-based retrieval should be applied for a given query. We show, for the first time, that by selecting between cluster-based retrieval and document-based retrieval for different queries, improved retrieval effectiveness can be achieved than using either type of retrieval for all queries.

The rest of the paper is organized as follows. We first discuss the related work in cluster-based retrieval in section 2, and provide the performance of optimal cluster retrieval in section 3. In section 4, we show the actual retrieval performance of a state-of-the-art retrieval technique on retrieving optimal clusters. An analysis of why optimal clusters are not retrieved at top ranks is given. We then describe the proposed technique in section 5. Empirical results are discussed in section 6. Section 7 concludes and points out possible directions for future work.

2. RELATED WORK

The use of document clustering in experimental IR systems dates back to 1960s. It was initially proposed as a means for improving efficiency of the system by Rocchio [16] and was adopted by the SMART system [17]. Goffman [7] proposed the notion of conditional probability of relevance, suggesting that the probability of relevance of a document should depend on the relevance of other documents to the same query. In 1972, van Rijsbergen [19] proposed the *Cluster Hypothesis* which states that closely associated documents tend to be relevant to the same query. Jardine and van Rijsbergen [10] suggested that document clustering could be used to improve both the effectiveness and the efficiency of retrieval. They introduced the notion of an *optimal cluster* search. The task for the retrieval system is to retrieve one or more clusters in their entirety to a query, by matching the query against clusters of documents instead of individual documents and ranking clusters based on their similarity to the query. Any document from a cluster that is ranked higher is considered more likely to be relevant than any document from a cluster ranked lower on the list. They suggested that "optimal" clusters exist that, if retrieved, would yield very large improvements in effectiveness relative to document-based retrieval. An optimal cluster is considered optimal if, when retrieved, it would give the maximum possible value for a retrieval criterion out of all clusters. This approach has been most common for cluster-based retrieval.

Numerous studies have been carried out in order to examine the comparative effectiveness of cluster-based retrieval and standard document-based retrieval. In most early attempts the strategy has been to build a static clustering of the entire collection in advance, independent of the user's query, and clusters are retrieved based on how well their centroids match the query. A hierarchical clustering technique is typically used in these studies as the size of the collection used is small, and different strategies for matching the query against the document hierarchy generated by such clustering algorithm have been proposed, most notably a top-down or a bottom-up search and their variants [10, 21, 2, 22]. While early studies on small collections showed that cluster-based retrieval had the potential of outperforming document-based retrieval for precision-oriented searches [2, 10], other experimental work [5, 22, 8] has suggested that document-based retrieval is generally more effective.

In 1980s, query-specific clustering has been proposed [25] which is to be performed on the set of documents retrieved by an IR system on a query. Willet [25] compared the efficiency and retrieval effectiveness of query-specific clustering to that of static clustering. He noted that substantial efficiency gains can be obtained with query-specific clustering since only a relatively small subset of the collection needs to be clustered. Retrieval experiments also showed that the effectiveness of both approaches are comparable. Willett [25] and Voorhees [22] experimented with different collections and showed that cluster-based retrieval did not outperform document-based retrieval, except on the small Cranfield collection that has been used in most early studies. Voorhees [22] concluded that the extent to which the cluster hypothesis hold on a collection seemed to have little effect on how well cluster-based retrieval performed as compared to document-based retrieval.

Hearst and Pedersen [9] and Tombros et.al. [18] examined the cluster hypothesis under the light of query-specific clustering. Both studies confirmed that the cluster hypothesis held for query-specific clustering, and showed that there existed an optimal cluster and that, if the IR system were able to retrieve that cluster, it would always perform better than with the document-based retrieval (e.g. SMART). Tombros et.al. [18] also showed that the number of top-ranked documents used for query-specific clustering does not have significant impact on clustering effectiveness, and query-specific clustering significantly outperformed static clustering for all experimental conditions. However, neither of the two studies addressed the question of if and how optimal clusters could be identified or used automatically in retrieval without relevance judgments. Instead, the quality of clusters were determined manually by users or based on the number of known relevant documents they contain.

Griffiths et al. [8] reported on a comparative study of cluster-based retrieval using several hierarchic document clustering methods. They found that methods which gave good retrieval results all yielded large number of small clusters. This finding led the authors to using the nearest neighbor clusters for retrieval. The experiments showed that this very simple, overlapping clustering method could give a level of retrieval effectiveness comparable with the other methods. However the best performance achieved was not better than document-based retrieval. By performing an analysis of the actual documents retrieved by both types of retrieval, the authors found that the relevant documents identified by the two types of retrieval were rather different. They reasoned that for a retrieval system to handle a wide range of possible queries, it would be ideal if the system contains both kinds of retrieval mechanisms because a system could switch to cluster-based retrieval if document-based retrieval proves to be unsatisfactory. They tried to combine the outputs from the two types of retrieval into a single set of documents but found that it would be more successful just to select one strategy to provide the output.

Croft and Thompson [4] described an adaptive mechanism that tries to learn which retrieval strategy is most appropriate for a given query, including choice between cluster and non-cluster searches. While the approach had some merit, the experiments showed that it was less successful than consistent use of just a single strategy.

There has been resurgence of cluster-based retrieval in the past few years. The main spirit is to use clusters as a form of document smoothing. Topic models are constructed from clusters and documents are smoothed with these topic models, to improve document retrieval [13, 11, 27]. While these methods showed that clusters can indeed improve retrieval performance automatically on modern test collections, their goal is still to directly retrieve documents. This somewhat deviates from the original spirit of cluster-based retrieval which is to find the best clusters or groups of documents.

The originality of our work consists in a detailed analysis of the reasons why optimal clusters are often not retrieved at top ranks with state-of-the-art retrieval techniques, and a selection mechanism that would allow the system to select which strategy (document or cluster-based retrieval) should be applied to a given query.

3. PERFORMANCE OF OPTIMAL CLUSTERS

Table 1. Statistics of data sets

Collection	Contents	# of Docs	Size	Average # of Words/Doc[1]	Queries	# of Queries with Relevant Docs
WSJ	Wall Street Journal 1987-92	173,252	0.51 Gb	465.8	TREC topics 51-100 & 151-200 (title only)	100
LA	LA Times	131,896	0.48 Gb	526.5	TREC topics 301-400 (title only)	98
TREC45	TREC disks 4 & 5 - The Financial Times, 1991-94; Federal Register, 1994; Congressional Record, 1993; Foreign Broadcast Information Service (FBIS); The LA Times.	556,077	2.14 Gb	541.9	TREC topics 301-400 (title only)	100

Table 2. Optimal cluster performance (optimal clusters are identified using document relevance judgment)

Collection	Metric	Document–based Retrieval	Cluster –based Retrieval
TREC45	Prec. at 5 docs	0.4140	0.8540
	Mean avg. prec.	0.2011	0.4317
	Avg. # of opti. clus.	-	52
WSJ	Prec. at 5 docs	0.5060	0.8800
	Mean avg. prec.	0.2958	0.5054
	Avg. # of opti. clus.	-	57

In this section, we investigate optimal cluster retrieval performance on TREC test collections using standard precision-recall measures. We first perform document-based retrieval using the query likelihood retrieval model [15, 14]. Next, we take the top 1000 retrieved documents and cluster them using the K Nearest Neighbor clustering method [6]. We set K to be 5. These decisions are made because previous work have shown that query-specific clustering gives at least as good a performance as static clustering while being more efficient [25], that the number of top-ranked documents used for query-specific clustering does not have significant impact on clustering effectiveness [18], and that the clustering method that gives best results typically results in a large number of small clusters [8]. The cosine measure is used to determine the similarity between documents. We score the clusters by counting the number of known relevant documents they contain and rank them in descending order of the scores. The final retrieval output is formed by displaying all the documents from a cluster that is ranked higher before the documents from a cluster that is lower on the list. Documents from the same cluster are ordered by their closeness to the query. We are most interested in precision at high levels (i.e. precision at docs) because it directly indicates how well an optimal cluster can perform if it is ranked at the top which is the goal of cluster-based retrieval. The data sets are TREC topics 301-400 (title only) against the whole disks of TREC disk 4 and 5 (TREC45), and TREC topics 51-100 and 151-200 (title only) against the Wall Street Journal (WSJ) collections. Both the queries and collections have been stemmed and stopped using the INQUERY list of 418 words. The statistics of the data sets are given in table 1. The performance of document-based retrieval and optimal cluster retrieval are given in table 2. For each data set, we show precision at 5, mean average precision, and the average number of optimal clusters per query. Here, we take optimal clusters to be those that give a precision that is better than document-based retrieval with the same number of documents (as that in each cluster) taken from the top of the retrieved list. That is, if a cluster has K documents, we take the same number of top retrieved documents from document-based retrieval, and compare their performance to that of the documents from the cluster. If the performance of the cluster is better then the cluster is considered optimal. The only exception is when the top K retrieved documents from document-based retrieval are all relevant. In this case optimal clusters are those that will produce the same level of performance, which means all the documents in them are relevant. From table 2, we observe that, on average, there are a good number of optimal clusters per query. If optimal clusters can be identified, the retrieval performance can be significantly higher than document-based retrieval. It should be noted, however, that this performance can only be reached if a retrieval strategy infallibly selects the best cluster for each query, and produces the perfect ranking.

[1] This is calculated when no stop words were removed and no stemming was performed.

4. RETRIEVAL OF OPTIMAL CLUSTERS

We have examined the best possible performance that optimal cluster retrieval could give, now we investigate what the actual performance is in retrieval of optimal clusters using a state-of-the-art retrieval technique. The query likelihood model has shown to be simple yet effective for document-based retrieval [3]. We concatenate documents in the same cluster and treat each cluster as if it were a big document. The clusters are ranked by the probability of how likely the cluster would have generated the query - *P(Q|Cluster)*.

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

Table 3. Cluster-based retrieval with query likelihood (QL).

Collection	Metric	Document Retrieval	Cluster Retrieval
TREC45	Prec. at 5 docs	0.4140	0.3240
	Mean avg. prec.	0.2011	0.1580
	MRR	-	0.5514
WSJ	Prec. at 5 docs	0.5060	0.4520
	Mean avg. prec.	0.2958	0.2262
	MRR.	-	0.5640

where $P(q_i|Cluster)$ is specified by the cluster language model

Table 4. Analysis of Query 306, "*" means relevant documents.

Cluster Rank	Num. of Rel. Docs	Cluster ID	Member Docs	Cluster/Doc. Log QL	Freq. of Term "death"	Freq. of Term "civilian"	Freq. of Term "africa"	Cluster/Doc. Length in Terms	Num. of Unique Terms
1	1	C636	-	-15.436329	12	12	141	3735	1170
			*CR93H-2896	-16.520723	1	12	107	1905	549
			FR940712-2-00058	-20.346148	3	0	9	315	149
			FT941-12410	-20.981014	4	0	6	563	328
			CR93E-250	-20.996843	2	0	13	627	377
			FR940712-2-00057	-21.119274	2	0	6	325	187
3	3	C2	-	-15.550194	87	4	18	2344	691
			*FBIS3-25118	-16.796564	20	2	4	548	263
			*FBIS3-471	-17.601721	26	1	3	635	313
			FBIS4-24155	-18.422461	11	1	1	207	143
			*FBIS3-602	-19.521227	25	0	6	739	356
			FBIS3-470	-20.388792	5	0	4	215	136
4	0	C208	-	-15.615210	17	2	77	1573	709
			FBIS4-24155	-18.422461	11	1	1	207	143
			FBIS4-48773	-19.985443	0	1	10	128	81
			FBIS4-23488	-20.339703	3	0	5	91	57
			CR93E-2102	-20.836098	1	0	58	1073	524
			FBIS4-1186	-21.120800	2	0	3	74	51
10	1	C886	-	-15.889583	10	3	23	811	480
			FBIS3-940	-19.500814	1	1	3	119	89
			*LA112089-0041	-20.113148	4	0	5	106	79
			FT932-13820	-20.212637	1	2	1	309	245
			LA060389-0047	-20.225418	2	0	12	226	140
			LA102190-0150	-21.403337	2	0	2	51	40
13	5	C14	-	-15.940542	9	15	17	1732	750
			*FT942-7623	-17.831032	1	6	8	531	319
			*FBIS4-28901	-18.292290	5	2	2	329	224
			*FBIS4-23790	-18.964863	2	2	3	409	262
			*FT942-9707	-19.172813	1	2	2	107	91
			*FBIS4-23738	-20.970539	0	3	2	356	233
77	5	C80	-	-16.665789	9	16	13	2258	891
			*FT942-7623	-17.831032	1	6	8	531	319
			*FT942-15976	-19.275204	3	1	2	269	187
			*FBIS4-47810	-19.629265	1	7	1	616	277
			*FT943-15255	-20.335827	3	1	1	511	340
			*FBIS4-912	-20.905777	1	1	1	331	226

Table 5. Analysis of Query 301.

Cluster Rank	Num. of Rel. Docs	Cluster ID	Member Docs	Cluster/Doc. Log QL	Freq. of Term "organize"	Freq. of Term "crime"	Freq. of Term "international"	Cluster/Doc. Length in Terms	Num. of Unique Terms
1	2	C72	-	-12.652414	86	148	23	3607	1384
			*CR93E-9750	-12.874592	59	56	16	1833	805
			FBIS4-41991	-14.538500	8	34	2	593	337
			*FBIS3-23986	-15.216642	14	21	5	599	380
			FBIS4-40936	-16.797482	5	15	0	319	192
			LA091290-0022	-18.363014	0	22	0	263	199
61	0	C27	-	-14.331124	59	32	14	2672	602
			FBIS4-33483	-16.174446	13	9	2	650	306
			FBIS4-33498	-16.174446	13	9	2	650	306
			FBIS4-36212	-16.174446	13	9	2	650	306
			FBIS4-24890	-17.15856	1	5	2	207	137
			FBIS4-22374	-18.218878	19	0	6	515	284
162	5	C68	-	-14.899220	35	30	6	1862	623
			*FBIS4-45477	-15.935877	11	10	2	496	308
			*FBIS4-31645	-16.754736	8	6	1	342	230
			*FBIS4-45469	-16.765894	8	6	1	347	231
			*FBIS3-49567	-17.730715	4	4	1	333	215
			*FBIS3-60093	-17.755371	4	4	1	344	218

$$P(w \mid Cluster) = \lambda P_{ML}(w \mid Cluster) + (1-\lambda) P_{ML}(w \mid Coll)$$
$$= \lambda \frac{tf(w, Cluster)}{\sum_{w' \in cluster} tf(w', Cluster)} + (1-\lambda) \frac{tf(w, Coll)}{\sum_{w' \in V} tf(w', Coll)} \quad (2)$$

where $P_{ML}(w|Cluster)$ is the maximum likelihood estimate of word w in the cluster, $P_{ML}(w|Coll)$ is the maximum likelihood estimate of word w in the collection, $tf(w, Cluster)$ is the number of times w occurs in the cluster, V is the vocabulary, $tf(w, Coll)$ is the number of times w occurs in the entire collection, and λ is a general symbol for smoothing. λ takes different forms when different smoothing methods are used. For example, for Jelinek-Mercer smoothing, λ is simply an arbitrary weight between 0 and 1; for Bayesian smoothing with the Dirichlet prior, λ takes the form

$$\lambda = \frac{\sum_{w' \in D} tf(w', D)}{\sum_{w' \in D} tf(w', D) + \mu} \quad (3)$$

where w' is any word, $tf(w', Cluster)$ is the number of times w' occurs in the document D, and μ is the Dirichlet smoothing parameter. In our experiments, we use Dirichlet smoothing with parameter set to 1000.

The results are shown in table 3. Similar to section 3, we report on precision at top 5 documents and mean average precision. In addition, we evaluate how well the system ranks the optimal clusters using the mean reciprocal rank measure (MRR) [28]. We form a cluster relevance judgment set by taking the optimal clusters identified for each query from the analysis performed in section 3. We go through the list of ranked clusters and mark the highest rank at which an optimal cluster is retrieved. The reciprocal of the rank is computed. The MRR score is the average of reciprocal ranks across all queries. The evaluation is appropriate because retrieving optimal clusters in the top ranks is the goal of any optimal cluster retrieval system. The results confirm with previous studies [8, 22] that the task of retrieving optimal clusters is very hard, and despite the fact that there are a decent number of optimal clusters per query, those clusters are typically not retrieved at the top rank. The overall performance of retrieving clusters is inferior to that of retrieving documents. To find out why this is the case, we performed the analysis presented in tables 4 and 5. We show part of the ranked cluster list for query 301 (table 5) and 306 (table4). The actual queries are shown in figure 1. The narrative field is taken from the corresponding TREC topic to show how relevance is judged for documents. For each cluster on the ranked list, the rank at which the cluster is retrieved, the number of relevant documents in the cluster, the cluster ID, member documents in the cluster, and the respective query likelihood of the cluster and member documents (the log of the query likelihood scores are shown) are given in the first five columns. In the next three columns, the table gives frequencies of each query term in the cluster as a whole and in individual documents in the cluster. The last two columns show the cluster and document length in terms of the number of indexing terms, as well as the number of unique terms contained within the cluster and each member document. The member documents of a cluster that are relevant are marked with an "*" in front the document ID in the 4th column.

Let us take a close look at query 306. The query is "African civilian deaths". After stemming and stopword removal, the query becomes "africa civilian death". Six clusters on the ranked cluster list are shown. The optimal clusters for the query are C14 and C80. Both clusters have five relevant documents. However, simply based on the term occurrences the system was not able to assign high ranks to them. The top ranked cluster is C636 which has only one relevant document. We observe that the document that is relevant is very long with lots of occurrences of the query terms. Therefore, even if the other documents are not relevant and with only few infrequent occurrences of query terms, the overall query likelihood of the cluster would still be good. Cluster C2 is ranked the third on the list. The occurrences of the query terms spread more evenly across the member documents than C636. There are 3 documents that are judged relevant but by reading document FBIS4-24155 we found that it is very similar in content to another document that is judged relevant. It may have been misjudged or judged by a assessor with really strict criterion when the relevance judgment set is created. The next cluster on the list is C208 which has no relevant documents. We found that even though the individual documents may not have all query terms appearing, the overall cluster query likelihood is good because the cluster model picks up different query terms from different documents. For example, document CR93E-2102 is very long and it contains many occurrences on the query term "africa", thus this document contributes largely to the overall cluster frequencies of that term. The cluster frequencies of term "civilian" come from only two of the five documents. We also observe that the relevant document FBIS3-602 does not have term "civilian". Instead of using "civilian", the article makes use of more descriptive terms such as victims, teachers, women, and children involved in the incidents. The term "casualties" is often used instead of "death". This gives an example of the case when the language used in the document is quite different from the query language. The clusters C14 and C80, while optimal, have very low frequency counts on the query terms, thus their query likelihood score is lower than the other clusters. Across all clusters for this query we observe that the optimal clusters tend to have documents with query likelihood close to each other and also to the overall cluster query likelihood. Non-optimal clusters tend to have more documents with zero occurrences of one or more query terms. However, after examining several other queries, we found that this is not typically true. For example, for query 301 (table 5), cluster C27 does not have any relevant document but four out of the five member documents have all terms appearing.

Query 301: International Organized Crime
After stemming and stopping: international organize crime
Narrative:
A relevant document must as a minimum identify the organization and the type of illegal activity (e.g., Columbian cartel exporting cocaine). Vague references to international drug trade without identification of the organization(s) involved would not be relevant.

Query 306: African Civilian Deaths
After stemming and stopping: africa civilian death
Narrative:
A relevant document will contain specific casualty information for a given area, country, or region. It will cite numbers of civilian deaths caused directly or indirectly by armed conflict.

Figure 1. Query 301 and 306. Narrative is given for information on relevance judgment.

Table 6. Retrieval performance with proposed selection mechanism compared to using cluster-based retrieval consistently for all queries. Retrieval is done using query likelihood model. "*" means that there is a significant improvement in performance using Wilcoxon test with 95% confidence. The percentage of improvement in performance is given in parentheses.

Collection	Cluster QL			Proposed technique		
	Prec. at 5 docs	MAP	MRR	Prec. at 5 docs	MAP	MRR
WSJ	0.4520	0.2262	0.5640	0.5200* (+15.0%)	0.2981* (+31.8%)	0.6436* (+14.1%)
LA	0.2822	0.1973	0.5321	0.3567* (+26.4%)	0.2563* (+29.9%)	0.6170* (+15.9%)
TREC45	0.3240	0.1580	0.5514	0.4500* (+38.9%)	0.2063* (+30.6%)	0.6146* (+8.8%)

Based on these observations, we devised a new technique that would allow the system to decide on whether to use optimal cluster retrieval or document-based retrieval for a given query. We describe this approach in the next section.

5. PROPOSED TECHNIQUE

From the analysis performed in section 4, we can see that there are several aspects of optimal cluster retrieval that can be improved, including the cluster representations, ranking strategies, among others. In the scope of this paper, we propose a new technique that addresses some of the aspects by taking into consideration how well a cluster as well as its member documents matches the query. We have observed in the analysis that the member documents in optimal clusters tend to have similar query likelihood[2] to that of the cluster. The intuition is that the less the member document query likelihood varies from the cluster query likelihood, the more likely that the documents contribute evenly to the cluster model. Clusters with large variability of member document query likelihood from the cluster query likelihood may mean that only some member documents contribute largely to the cluster model (e.g. cluster C636 in table 4) or the individual documents contribute to different query term occurrences in the cluster model for the cluster query likelihood to be high but the documents tend to have low query likelihood (e.g. cluster C208 in table 4). A popular way to measure variation is variance [29]. It is computed as the average squared deviation of each number in a distribution from its mean. Taking a similar approach, we compute the average squared deviation of the query likelihood of each document in a cluster from the cluster query likelihood. That is,

$$WCD = \frac{\sum_{d \in C} (MS_d - MS_C)^2}{K} \qquad (4)$$

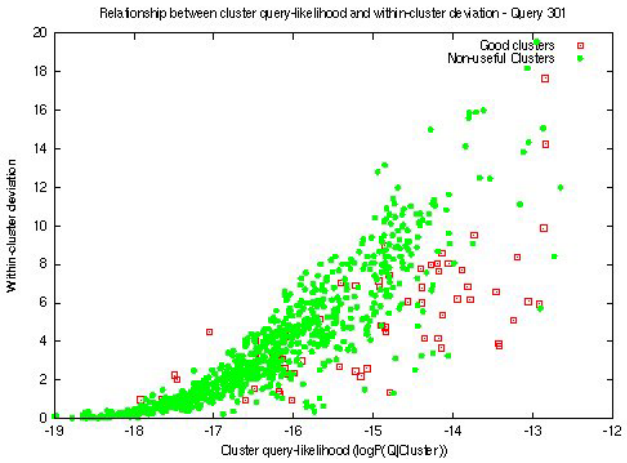
where C stands for a cluster, d stands for any document in the cluster, K is the number of documents in the cluster, MS_d is the a measure of closeness of the document to the query, and MS_C is a measure of closeness of the cluster to the query. In this work, we use query likelihood. We call this metric defined in equation (4) the query-informed within-cluster deviation. We conjecture that the good clusters would be those with high cluster query likelihood but low query-informed within-cluster deviation. If for a given query, such clusters exist then the system is likely to succeed with cluster-based retrieval. In this case, the system applies cluster-based retrieval and ranks these clusters before others. A document list is created by displaying documents from the first cluster, then those from the second cluster, and so on. Documents from the same cluster are ranked according to their closeness score to the query. If no clusters are considered satisfactory, the system outputs the list of documents produced from document-based retrieval. While query likelihood is used to measure the closeness of documents/clusters to the query in this paper, other techniques (e.g. relevance model) can be easily applied and combined with the proposed technique. We describe experiments with this technique in the next section.

6. EXPERIMENTS AND RESULTS

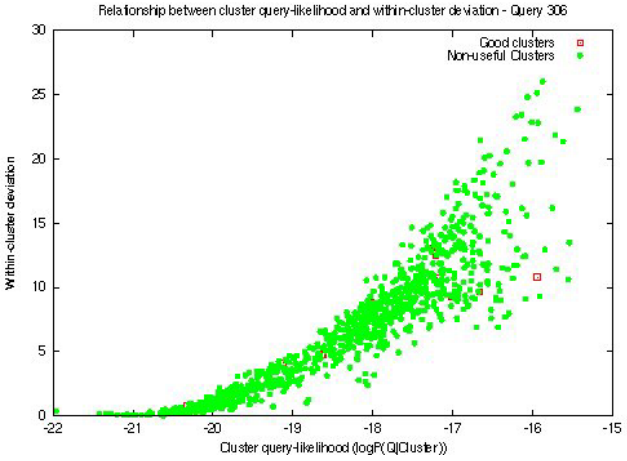
We present experimental results in this section. In all experiments, both the queries and documents are stemmed, and stopwords are removed using the standard INQUERY stoplist of 418 words. We use three data sets: TREC topics 51-100 and 151-200 (title only) on the WSJ collection, TREC topics 301-400 (title only) on the TREC45 collection, and TREC topics 301-400 (title only) on the LA collection. We select these data sets because WSJ and LA are examples of news collections and they are homogeneous in terms of both document size and topics. TREC45 is selected an example of mixed collection that contains Federal Register, Congressional Records, as well as some news collections. Similar to experiments in section 3, we use query-specific clustering with the K Nearest Neighbor method. K is set 5. The cosine measure is used to determine the similarity between documents and top 1000 documents from document-based retrieval are clustered.

In order to decide which clusters are potentially good clusters, we need to find a threshold for both the cluster query likelihood and the query-informed within-cluster deviation (WCD). There are two parameters to determine. A cluster is considered good if its cluster query likelihood falls into the upper x percent of its value range and its WCD falls into the lower y percent of its value range. The value ranges are determined from all clusters for the given query. The WSJ data set is selected as the training collection for determining these parameters. An exhaustive parameter search is applied and the best retrieval performance is obtained at parameters set to 80 (x for cluster query likelihood) and 40 (y for

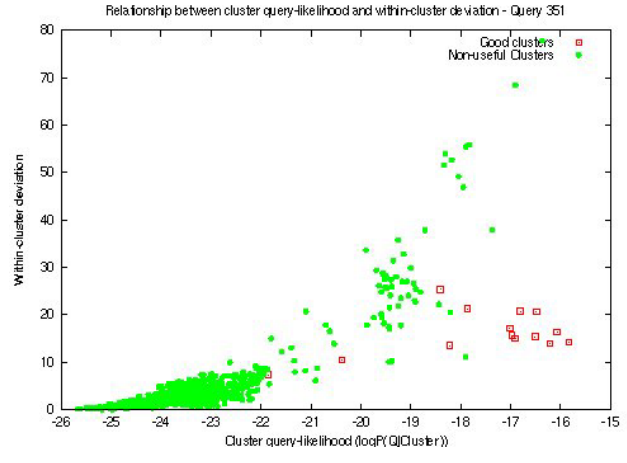
[2] Other matching function between query and documents/clusters can be used. We use query likelihood here for convenience of discussion.



(a)



(b)



(c)

Figure 2. Example results from the proposed technique

WCD). We then apply these parameters to the TREC45 and LA data sets. If the system finds clusters that satisfy the requirement on the cluster query likelihood and WCD, the system performs cluster-based retrieval. If no such clusters are found, document-based retrieval is used. Retrieval results are reported in table 6 and table 7. Table 6 shows the performance of using cluster-based retrieval for all queries and our selection technique. Table 7 compares using document-based retrieval for all queries and the proposed technique. "*" means that there is a significant improvement in effectiveness of our technique over that of the baselines with the Wilcoxon test at 95% confidence. From the tables, we can see that there are significant improvements of our selection mechanism over consistent use of just document-based retrieval or cluster-based retrieval, both for precision at top 5 documents retrieved, mean average precision. There are 28, 36, and 33 queries that used cluster-based retrieval, on WSJ, TREC45, and LA respectively. We also evaluated the MRR measures of baseline cluster-based retrieval and our technique for those queries that used cluster retrieval. Again, significant improvements are observed. We plot the relationship between the log query likelihood of clusters and their WCD for query 301, 306, and 351 on TREC45 data set in figure 2. The small box-shaped points stand for the clusters that are identified in our evaluation that have better performance if ranked at the top respective to document-based retrieval at the same cutoff. The dots represent clusters that are not helpful for retrieval. Plot (a) is an example case when cluster-based retrieval is applied which improves retrieval performance. Plot (b) gives an example when good clusters are not separable from non-useful clusters. Our system is able to decide not to use cluster-based retrieval. Plot (c) shows the case that doing cluster-based retrieval is as good as document-based retrieval.

Table 7. Retrieval performance with proposed selection mechanism compared to using document-based retrieval consistently for all queries. Retrieval is done using query likelihood model. "*" means that there is a significant improvement in performance using Wilcoxon test with 95% confidence. The percentage of improvement in performance is given in parentheses.

Collection	Doc QL		Proposed technique	
	Prec. at 5 docs	MAP	Prec. at 5 docs	MAP
WSJ	0.5060	0.2958	0.5200* (+2.8%)	0.2981* (+0.8%)
LA	0.3367	0.2468	0.3567* (+5.9%)	0.2563* (+3.8%)
TREC45	0.4140	0.2011	0.4500* (+8.7%)	0.2063* (+2.6%)

7. CONCLUSIONS AND FUTURE WORK

In this paper, we examined the optimal retrieval cluster performance on TREC test collections and provided a detailed analysis as to why optimal cluster are not retrieved at top ranks. The analysis shows that there are several possible reasons. One is that the representation of a cluster created by simply combining the member documents may not be best suitable for cluster-based retrieval. Also, the current cluster-based retrieval techniques do not take into consideration of the how well the member documents match the query. We proposed a query-informed within-cluster deviation measure and a selection mechanism based on this measure, and showed that the system is able to automatically decide whether to use cluster-based retrieval for a given query. Our technique addresses some aspects of the identified problems and we plan to investigate other aspects in our future work. We plan to examine different ways of constructing cluster

representations and retrieval models that are more suitable for cluster-based retrieval.

8. ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF grant #CNS-0454018 . Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect those of the sponsor.

9. REFERENCES

[1] Croft, W. B. (1978). *Organizing and Searching Large Files of Document Descriptions*. Ph.D. dissertation, University of Cambridge.

[2] Croft, W. B. (1980). A model of cluster searching based on classification. *Information Systems*, Vol. 5, pp. 189-195.

[3] Croft W. B., & Lafferty, J (eds.) (2003). *Language Modeling for Information Retrieval*. In Kluwer International Series on Information Retrieval, Volume 13, Kluwer Academic Publishers.

[4] Croft, W. B., & Thompson, R. H. (1984). The use of adaptive mechanisms for selection of search strategies in document retrieval systems. In: van Rijsbergen, C. J, (Ed.) *Research and Development in Information Retrieval*. Cambridge, UK: Cambridge University Press.

[5] El-Hamdouchi, A. & Willet, P. (1989). Comparison of hierarchic agglomerative clustering methods for document retrieval. *The Computer Journal*, 32(3), pp. 220-227.

[6] Yang, Y, & Liu, X. (1999). A re-examination of text categorization methods. In *SIGIR-99*.

[7] Goffman, W. (1969). An indirect method of information retrieval. *Information Storage and Retrieval*, 4, pp. 361-373.

[8] Griffiths, A., Luckhurst, H.C., and Willett, P. (1986). Using interdocument similarity information in document retrieval systems. *Journal of the American Society for Information Science*, 37, pp. 3-11.

[9] Hearst, M.A., and Pedersen, J.O. (1996). Re-examining the cluster hypothesis: Scatter/Gather on retrieval results. In *SIGIR 1996*, pp. 76-84.

[10] Jardine, N. and van Rijsbergen, C.J. (1971). The use of hierarchical clustering in information retrieval. *Information Storage and Retrieval*, 7:217-240.

[11] Kurland, O. and Lee, L. (2004). Corpus structure, language models, and ad hoc information retrieval. In *Proceedings of SIGIR'04 conference*, pp. 194-201.

[12] Leuski, Anton. (2001). Evaluating Document Clustering for Interactive Information Retrieval. In *Proceedings of CIKM'01 conference*, pp.33-40.

[13] Liu, X. and Croft, W. B. (2004). Cluster-based retrieval using language models. In *Proceedings of SIGIR'04 conference*, pp. 186-193.

[14] Miller, D., Leek, T., and Schwartz, R. (1999). A hidden Markov model information retrieval system. In *SIGIR 1999*, pp. 214-221.

[15] Ponte, J., and Croft, W.B. (1998). A language modeling approach to inormation retrieval. In *SIGIR 1998*, pp.275-281.

[16] Rocchio, J. J. (1966). *Document Retrieval Systems – Optimization and Evaluation*. Ph. D. thesis, Harvard University.

[17] Salton, G. (1971). Cluster search strategies and optimaization of retrieval effectiveness. In G. Salton, editor, *The SMART Retrieval System*, pp. 223-242. PrenticeHall, Englewood Cliffs, N. J..

[18] Tombros, A.; Villa, R.; and Van Rijsbergen, C.J. (2002). The effectiveness of query-specific hierarchic clustering in information retrieval, *Information Processing and Management*, 38, pp. 559-582.

[19] van Rijsbergen, C.J. (1972). *Automatic Information Structuring and Retrieval*. Ph.D. thesis, University of Cambridge.

[20] van Rijsbergen, C. J. (1974). Foundation of evaluation. *Journal of Documentation*, 30, pp. 365-373.

[21] van Rijsbergen, C.J. & Croft, W. B. (1975). Document clustering: An evaluation of some experiments with the Cranfield 1400 collection. *Information Processing & Management*, 11, pp. 171-182.

[22] Voorhees, E.M. (1985). The cluster hypothesis revisited. In *SIGIR 1985*, pp.188-196.

[23] Voorhees, E. M. (1985). *The effectivenss and efficiency of agglomerative hierarchic clustering in document retrieval*. Ph.D. Thesis, Technical Report TR 85-705 of the Department of Computing Science, Cornell University.

[24] Willet, P. (1988). Recent trends in hierarchical document clustering: A critical review. *Information Processing & Management*, 24(5):577-597.

[25] Willet, P. (1985). Query specific automatic document classification. *International Forum on Information and Documentation*, 10(2), pp. 28-32.

[26] Xu, J., and Croft, W.B. (1999). Cluster-based language models for distributed retrieval. In *SIGIR 1999*, pp.254-261.

[27] Zhang, Y., Callan, J., & Minka, T. (2002). Novelty and redundancy detection in adaptive filtering. In *SIGIR 2002*, pp. 81-88.

[28] Voorhees E. M. (1999). The TREC-8 Question Answering Track Report. In *Proceedings of TREC-8*.

[29] Mukhopadhyay, N. (2000). *Probability and Statistical Inference*, Marcel Dekker Inc.