Evaluating Text Representations for Retrieval of the Best Group of Documents

Xiaoyong Liu and W. Bruce Croft
CIIR, Computer Science Department, 140 Governors Drive,
University of Massachusetts, Amherst, MA 01003, USA
{xliu, croft}@cs.umass.edu

Abstract. Cluster retrieval assumes that the probability of relevance of a document should depend on the relevance of other similar documents to the same query. The goal is to find the best *group* of documents. Many studies have examined the effectiveness of this approach, by employing different retrieval methods or clustering algorithms, but few have investigated text representations. This paper revisits the problem of retrieving the best group of documents, from the language-modeling perspective. We analyze the advantages and disadvantages of a range of representation techniques, derive features that characterize the good document groups, and experiment with a new probabilistic representation as a first step toward incorporating these features. Empirical evaluation demonstrates that the relationship between documents can be leveraged in retrieval when a good representation technique is available, and that retrieving the best group of documents can be more effective than retrieving individual documents.

Keywords: Text Representation, Document Retrieval, Cluster Retrieval, Cluster Representation, Representation Techniques.

1 Introduction

The standard approach to document retrieval has been based on the Probability Ranking Principle [13]. It assumes that the relevance of documents could be assessed independently. The fact that a document is relevant does not contribute to predicting the relevance of a closely-related document. Cluster retrieval, on the other hand, assumes that the probability of relevance of a document should depend on the relevance of other similar documents to the same query [17]. Document groups are usually formed by utilizing some clustering algorithms, and the system's goal is to find the best *group* of documents [4]. Jardine and van Rijsbergen, and others [4, 3, 15] studied the performance of the ideal retrieval strategy that infallibly finds the best group (they call it an "optimal" cluster), and showed that effectiveness would be far better than a search based on individual documents.

Many studies have examined the effectiveness of cluster retrieval, by employing different retrieval methods or clustering algorithms [1, 2, 4, 7, 15, 16, 18, 20]. The findings have been inconclusive as to whether a real retrieval strategy is able to retrieve the good document groups in the top ranks. Except for precision-oriented searches on very small data sets [1, 4], retrieving individual documents is found to be generally more effective [2, 18, 20, 8]. However, most studies represented document groups either by concatenating the documents within each group [1, 6, 8] or a centroid

vector [18], and only a couple of studies [7, 10] have compared different representations. The number of representations examined is small. There has been a resurgence of research in cluster-based retrieval in the past few years [8, 6, 14]. The general approach is to use clusters as a form of document smoothing. The IR system's goal is still directly ranking individual documents, not clusters. The issue of how to identify good document groups remains unaddressed. In this paper, we revisit the problem of retrieving the best group of documents, from the language-modeling perspective. We aim to provide an extensive evaluation of existing and new representation techniques. We argue that whether good document groups could be successfully identified by an IR system largely depends on how they are represented.

In this work, document groups are generated by a clustering algorithm. It is possible to have other types of document groups (e.g. user-generated discussion groups) in other applications. For simplicity of discussion and to avoid possible confusion in this paper, we use "cluster" and "document group" interchangeably. We distinguish between cluster retrieval and cluster smoothing. Cluster retrieval directly ranks groups of documents (clusters) whereas cluster smoothing ranks documents but uses clusters to smooth the document probabilities. We will use "good cluster" instead of "optimal cluster" or "the best groups of documents" in our discussions. An optimal cluster is one that gives the best precision out of all clusters [4] and a good cluster is a relaxed definition of optimal cluster. It refers to any cluster that gives better precision than document retrieval with the same cutoff from the top of the result list.

2 Cluster Representations

To use the language modeling (LM) approach for retrieving clusters, we first need to derive language models from cluster representations and then apply retrieval models. Let's take the query likelihood (QL) retrieval model for example. Clusters are ranked based on their likelihood of generating the query, i.e. $P(Q|Cluster)$. It can be estimated by equation (1) where Q is the query, q_i is the ith term in the query, and $P(q_i|Cluster)$ is the cluster language model (computed using equation (2)). $P_{ML}(w|Cluster)$ and $P_{ML}(w|Coll)$ are the maximum likelihood estimates of word w in the document and the collection, $tf(w, Cluster)$ and $tf(w, Coll)$ are the term frequencies of w in the cluster and the collection, V is the vocabulary, and $\bullet$ is a general symbol for smoothing which takes different forms when different smoothing methods are used [8].

1. Concatenating documents. The standard approach to representing clusters is to treat them as if they were big documents formed by concatenating their member documents. Thus, $tf(w, Cluster)$ is computed by equation (3) where $Cluster = \{D_1,..., D_k\}$ and k is the number of documents in a cluster. Clusters are ranked by equation (1) with components estimated from equations (2) and (3).

This representation, while being simple and intuitive, may have a number of problems. For example, if cluster A has a document that is very long and has many occurrences of the query terms while other member documents are short with only few query terms appearing, then simply concatenating these documents would result in a representation that is largely biased by one particular document [9]. In contrast, a cluster B has more relevant documents but do not have as many occurrences of the query terms when combined. Cluster A will be ranked higher because of the

probability estimates. This is what we want to avoid because the quality of clusters is usually judged by the total number of relevant documents they contain rather than how good one of the documents is [15]. Clusters with more relevant documents are considered better. The problem with this representation is that the differences of query term frequencies in documents with a higher QL mask the differences in term frequencies in the documents with a lower QL. A lesson learned from this is that a good representation should offset the bias toward documents with a higher QL, and one way to achieve this is to put more emphasis on documents with a lower QL.

2. Centroid vector. Clusters can also be represented by a centroid vector, or the document that is the most similar to the actual centroid, as in e.g. [7]. The representation can be formulated as equation (4). Clusters are ranked by equation (1) with components estimated from equations (2) and (4). Similar to concatenating documents, this method may also suffer from bias introduced by some member documents. It is possible that each member document only contributes largely to the estimate associated with one query term but different document contributes to different terms. So even if the individual documents are not relevant, the centroid vector may still look good.

Figure 1. Lanugage model formulations for different representations.

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

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

$$tf(w, Cluster) = \sum_{i=1}^{k} tf(w, D_i) \tag{3}$$

$$tf(w, Cluster) = \frac{\sum_{i=1}^{k} tf(w, D_i)}{k} \tag{4}$$

$$\begin{aligned} P(Q \mid Cluster) &= \max_{D_i \in Cluster} P(Q \mid D_i) \\ &= \max_{D_i \in Cluster} [\lambda P_{ML}(w \mid D_i) + (1-\lambda) P_{ML}(w \mid Coll)] \end{aligned} \tag{5}$$

$$\begin{aligned} P(Q \mid Cluster) &= \min_{D_i \in Cluster} P(Q \mid D_i) \\ &= \min_{D_i \in Cluster} [\lambda P_{ML}(w \mid D_i) + (1-\lambda) P_{ML}(w \mid Coll)] \end{aligned} \tag{6}$$

$$tf(w, Cluster) = \sum_{i=1}^{k} (\alpha_i * tf(w, D_i)) \qquad \text{where} \quad \sum_{i=1}^{k} \alpha_i = 1 \tag{7}$$

$$P(w \mid Cluster) = \sum_{i=1}^{k} [\beta_i * (\lambda P_{ML}(w \mid D_i) + (1-\lambda) P_{ML}(w \mid Coll))] \quad \text{where} \sum_{i=1}^{k} \beta_i = 1 \tag{8}$$

$$P(w \mid Cluster) = \left(\prod_{0 \le i \le k} P(w \mid D_i) \right)^{\frac{1}{k}} \tag{9}$$

3. Best document. [7] used the highest ranked document (e.g. by QL model in document retrieval) in a cluster as the representative. The hypothesis is that if this document is non-relevant then the rest of the cluster is very likely non-relevant. Clusters are ranked according to equation (5). The problem with this approach is not difficult to see with an example. Suppose we have two clusters, one with five relevant documents and the other with one relevant and four non-relevant documents. If the relevant document in the second cluster has a better QL score than any of those in the first cluster, then the retrieval model will rank the second cluster higher. But in reality, the first cluster is better.

4. Worst document. The lowest ranked document in a cluster was also used as the cluster representative in [7]. The hypothesis is that if that document is relevant then it is very likely that the rest of the cluster is also relevant. Clusters are ranked by equation (6). Again, we illustrate the problem with an example. Suppose we have two clusters, one with five non-relevant documents and the other with four relevant and one non-relevant document. If the non-relevant document in the second cluster has a lower QL than any of the non-relevant documents in the first cluster, the retrieval model will rank the first cluster higher, but in fact the second cluster is better.

5. TF mixture. [10] proposes a weighted mixture of term frequencies from member documents for representation, i.e. equation (7), where • is a weighting parameter between 0 and 1. Clusters are ranked by equation (1) with components estimated from equations (2) and (7). α in equations (7) is estimated by the first-stage retrieval log QL score of each document divided by the sum of log QL scores of all member documents in a cluster. Note that the log QL scores are negative. Setting α this way penalizes clusters with documents that match the query poorly.

The advantage of this approach lies in that it explicitly considers the contribution of individual documents to the cluster model. The disadvantage is that the α weight is difficult to determine. The current way of setting the weight may not be optimal as the performance of this representation does not vary much from the centroid vector representation discussed earlier (see section 4). We have experimented with several other ways of determining the weight but have not found a setting that will perform better than document retrieval.

6. DM mixture. The second method proposed by [10] is to build language models for individual member documents and the cluster language model is a weighted mixture of these member document models, i.e. equation (8). Again, λ is a general symbol for smoothing, and • is a weighting parameter between 0 and 1. • is estimated in the same way as α for the TF mixture method. Clusters are ranked by equation (1) with components estimated from equation (8).

Similar to TF mixture, this method has the advantage of explicitly modeling contributions from member documents. But again, it suffers from the difficulty of setting the • weight. Empirically, using the current way of setting the weight, this representation performs slightly better than TF mixture (see section 4).

7. Geometric mean. As we can see from previous representations, especially concatenating documents and centroid vector, the problem with summing up or averaging the query term frequencies in member documents is that differences in term frequencies in documents with higher QL mask the differences in term frequencies in documents with lower QL. We analyzed the ideal and real results of cluster retrieval using the QL model and the representation of concatenating documents in [9]. We found that, despite that there are plenty of good clusters per query, those clusters are typically not retrieved in the top ranks. We further identified the following features

that characterize good clusters: : a) a cluster model with good query likelihood, b) member document models with good query likelihood, and c) low variability in document model estimates. The existing representations don't account for these features and thus often fail to assign top ranks to good clusters.

These observations suggest a non-linear rescaling of the individual documents' language model estimates before averaging over the cluster as a way of emphasizing the documents with low QL. We experimented with a new representation that is based on the geometric mean of document model estimates. It is formulated as equation (9). We first derive the member document models P(w|D) and compute their geometric mean. Clusters are ranked by equation (1) combined with (9). The geometric mean is equivalent to taking the log of individual documents' estimates, computing the arithmetic mean of the logs, and exponentiating back for the final geometric mean score. This representation has the desired effect of emphasizing estimates close to 0 (documents with low QL) while minimizing differences between larger estimates. There is no need for additional parameter tuning other than the smoothing parameters associated with the document models. Theoretically, the geometric mean estimates need to be renormalized so that they still qualify as probability estimates. We found in our experiments, however, that the normalization significantly increases the computer processing time while being less effective in ranking clusters than the un-normalized method. We evaluate this representation and present the results using the un-normalized geometric mean in section 4. Geometric mean has been used in the geometric mean average precision measure introduced in TREC 2004 Robust track to account for a similar phenomenon observed with evaluating topic sets that contain poorly performed topics [19].

3 Experimental Setup

The data sets used in the experiments and analysis come from the TREC collections: Wall Street Journal (WSJ) 1987-92 with topics 51-100, Associated Press newswire (AP) 1988-90 with topics 101-150, TREC disks 1 & 2 (TREC12) with topics 151-200, and TREC disks 4 & 5 (TREC45) with topics 301-400. The queries are taken from the "title" field of TREC topics. The query sets are determined such that different collections do not share the same queries. Both queries and documents are stemmed with K-stem [5], and stopwords are removed based on the standard INQUERY list of 418 words. The WSJ data set is used as the training collection if parameter tuning is needed.

We use query-specific clustering in this work. Document retrieval using the query likelihood retrieval model [12, 11] is first performed with Dirichlet smoothing at 1000. The top 1000 retrieved documents are then clustered using the K Nearest Neighbor method (KNN) [21]. K is set to 5 (i.e. each cluster has five documents). The cosine similarity measure is used to determine the similarity between documents. Once we have the clusters, we represent and rank them using one of the methods described in section 2. As we mentioned previously, for cluster retrieval, the system's goal is to retrieve the best group of documents. Theoretically, only one cluster should be displayed. However, since the system has a ranked list of clusters, it is also a common practice to display some or all of them. This work focuses on the top retrieved cluster and the precision at 5 documents (PREC-5) is used for evaluation.

4 Experimental Results

There are four experimental questions that we would like to address. The first question is to compare the performance of the geometric-mean representation with the performance of the standard approach of concatenating documents. The results are given in table 1. The percentage improvement is given in parentheses. We observe that there is a large difference in effectiveness between these two representations. The geometric-mean approach gives at least a 9.9% improvement on any evaluation set over the standard approach.

The second experimental question is to compare the performance of the geometric-mean representation with that of document retrieval. Table 2 shows the results for precision at 5, 10, 15, and 20 documents. We observe that, except for precision at 15 and 20 on the AP collection, the geometric-mean representation for cluster retrieval consistently outperforms document retrieval across different data sets and at varying precision levels. If we focus on the first retrieved cluster, large performance gain (over 9%) is obtained on both WSJ and TREC45 collections while smaller improvements are observed on AP and TREC12 collections.

Table 1. Comparing cluster representations: geometric mean and concatenating docs.

Collection	Prec. At 5 docs	
	Concatenating docs	Geometric mean
WSJ	0.4400	0.5040 (+ 14.5%)
AP	0.4040	0.4440 (+ 9.9 %)
TREC12	0.4360	0.6000 (+ 37.6 %)
TREC45	0.3240	0.4520 (+ 39.5 %)

Table 2. Comparing cluster (geometric-mean representation) and document retrieval.

Eval. Metric	WSJ		AP		TREC12		TREC45	
	Doc	Cluster	Doc	Cluster	Doc	Cluster	Doc	Cluster
Prec. @ 5	0.4600	0.5040 (+9.6%)	0.4240	0.4440 (+4.7%)	0.5920	0.6000 (+1.4%)	0.4140	0.4520 (+9.2%)
Prec. @ 10	0.4320	0.4760 (+10.2%)	0.4040	0.4080 (+1.0%)	0.5460	0.5960 (+9.2%)	0.3820	0.4060 (+6.3%)
Prec. @ 15	0.4173	0.4587 (+9.9%)	0.3867	0.3813 (-1.4%)	0.5427	0.5747 (+5.9%)	0.3553	0.3700 (+4.1%)
Prec. @ 20	0.3950	0.4350 (+10.1%)	0.3880	0.3780 (-2.6%)	0.5210	0.5450 (+4.6%)	0.3385	0.3410 (+0.7%)

In order to gain a better understanding as to why the new representation works better on some of the collections than the others, we analyzed the queries and the intermediate and final outputs of the system. We found that the geometric-mean approach works well for queries that have four or fewer index terms. All queries on the TREC45 collection have fewer than 5 index terms, so most of the queries benefited from cluster retrieval with only 9 out of 100 queries that were slightly hurt by this technique. For queries that are longer, however, the proposed representation seems to lose its advantage. One possible reason is that, for shorter queries, good clusters tend to have all query terms but for longer queries it is rarely the case. Both relevant and non-relevant documents contribute to only some of the query terms, and at often times good clusters can contain fewer unique query terms than bad clusters. As the geometric mean is based on a product of query term probabilities in documents and clusters, if a term doesn't occur in a cluster, its collection probability is used instead, which

will result in smaller overall probability estimate for that cluster. Good clusters can receive lower ranks because of this. This type of queries is also difficult for document retrieval due to similar problems. Shorter queries do not have this problem because there are at least some good clusters that contain all the query terms, and bad clusters will not have more unique query terms than them.

The next experimental question is the comparison of seven different cluster representations (described in section 2). The results are presented in table 3. We can see that the geometric-mean representation is consistently better than all others on all four data sets. DM mix, TF mix, and Centroid methods are very similar to each other in performance. Except for the geometric-mean method, the performance of all the other representations is typically lower than that of document retrieval. If we order the representations from best to worst, we have this list: Geometric mean, DM mix, TF mix, Centroid, Concatenating documents, Worst Doc, Best Doc. We noticed that some of the representations are not stable and can perform well on some data sets but badly on others. For example, TF mix outperforms document retrieval on WSJ but does poorly on TREC45. Best Doc performs poorly on WSJ and AP but gives one of the best results on TREC12 and TREC45. Centroid, TF mix, DM mix, and Concatenating documents all seem to perform poorly on TREC45. Compared to these, the geometric-mean approach seems to be most stable.

The last question is comparing the performance of cluster retrieval with cluster smoothing [8]. Cluster smoothing is implemented following [8] and with query-specific clusters (which is the same set of clusters for cluster retrieval). The results are shown in table 3. Cluster retrieval using the geometric mean representation is consistently better than cluster smoothing in retrieval effectiveness. The other representations are typically less effective than cluster smoothing.

Table 3. Comparing different cluster representations. Prec @ 5 is used for evaluation.

Coll.	Doc Ret.	Cluster Smoothing	Cluster Retrieval						
			Concat.	Best Doc	Worst Doc	Centroid	TF mix	DM mix	Geometric
WSJ	0.4600	0.4480	0.4400	0.3840	0.4080	0.4800	0.4800	0.4920	0.5040
AP	0.4240	0.4440	0.4040	0.3600	0.3760	0.3800	0.3860	0.4240	0.4440
TREC12	0.5920	0.5440	0.4360	0.5080	0.4680	0.4400	0.4180	0.4120	0.6000
TREC45	0.4140	0.4140	0.3240	0.4120	0.4060	0.2940	0.3020	0.3960	0.4520

5 Conclusions and Future Work

In this paper, we have revisited the problem of retrieving the best group of documents within the language modeling framework. We empirically evaluated and compared document retrieval, cluster smoothing, and cluster retrieval with seven different cluster representations, including a new approach based on geometric mean as a first step toward incorporating these features. Experimental results show that the geometric-mean representation is a relatively stable method, and performs consistently better than document retrieval, cluster smoothing, and cluster retrieval using other representations. This work demonstrates that, with a good representation method, we can leverage the relationship between documents, and the effectiveness of retrieving documents as a group can be consistently better than that of retrieving them individually,

especially in the top rank positions. This work is in progress and we plan to look into other features that are likely to benefit cluster retrieval as well as feature combination.

6 Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF grant #CNS-0454018 and #CCF-005575.

7 References

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

[2] 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.

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

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

[5] Krovetz, R. (1993). Viewing Morphology as an Inference Process, In *SIGIR 1993*, 191-203.

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

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

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

[9] Liu, X. and Croft, W. B. (2006). Cluster-based retrieval from a language-modeling perspective. In *the Doctoral Consortium of SIGIR'06 conference*. Abstract in *SIGIR'06 Proceedings* pp. 737-738.

[10] Liu, X. and Croft, W. B. (2006). Representing clusters for retrieval. In *Proceedings of SIGIR'06 conference*, pp. 671-672.

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

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

[13] Robertson, S.E. (1977). The probability ranking principle in IR, *Journal of Documentation*, 33, 294-304.

[14] Tao Tao, Xuanhui Wang, Qiaozhu Mei, ChengXiang Zhai. (2006). Language model information retrieval with document expansion. In *Proceedings of HLT/NAACL 2006*.

[15] 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.

[16] 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.

[17] van Rijsbergen, C. J. and Sparck Jones K. (1973). A test for the separation of relevant and non-relevant documents in experimental retrieval collections. *Journal of Documentation*, vol. 29, pp. 251-257.

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

[19] Voorhees, E. M. (2005). The TREC robust retrieval track. In SIGIR Forum vol. 39, No. 1.

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

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