In the Proceedings of ACM SIGIR '96, August, 1996, Zurich

Reexamining the Cluster Hypothesis: Scatter/Gather on Retrieval Results [*]

Marti A. Hearst and Jan O. Pedersen
Xerox Palo Alto Research Center
3333 Coyote Hill Rd
Palo Alto, CA 94304
hearst,pedersen@parc.xerox.com

Abstract

We present Scatter/Gather, a cluster-based document browsing method, as an alternative to ranked titles for the organization and viewing of retrieval results. We systematically evaluate Scatter/Gather in this context and find significant improvements over similarity search ranking alone. This result provides evidence validating the cluster hypothesis which states that relevant documents tend to be more similar to each other than to non-relevant documents. We describe a system employing Scatter/Gather and demonstrate that users are able to use this system close to its full potential.

1 Introduction

An important service offered by an information access system is the organization of retrieval results. Conventional systems rank results based on an automatic assessment of relevance to the query [20]. Alternatives include graphical displays of interdocument similarity (e.g., [1, 22, 7]), relationship to fixed attributes (e.g., [21, 14]), and query term distribution patterns (e.g., [12]). In this paper we will discuss and evaluate the use of *Scatter/Gather* [4, 5] as a tool for navigating retrieval results.

The Scatter/Gather browsing paradigm clusters documents into topically-coherent groups, and presents descriptive textual summaries to the user. The summaries consist of topical terms that characterize each cluster generally, and a number of typical titles that sample the contents of the cluster. Informed by the summaries, the user may select clusters, forming a subcollection, for iterative examination. The clustering and reclustering is done on-the-fly, so that different topics are seen depending on the subcollection clustered.[1]

Scatter/Gather may be applied to an entire corpora, in which case static off-line computations may be exploited to speed dynamic on-line clustering [5]. We have recently shown [18] that this use of Scatter/Gather successfully conveys some of the content and structure of the corpus. However, that study also showed that Scatter/Gather was less effective than a standard similarity search when the subjects were provided with a query. That is, subjects constrained to only navigate a hierarchical structure of clusters that covers the entire collection were less able to find documents relevant to the supplied query than subjects who employed similarity search to focus interest on a sharp subset of the corpus.

It is possible to integrate Scatter/Gather with conventional search technology by applying it after a search to organize and navigate the retrieved documents, which then form the target document collection. The topic-coherent clusters can be used in several ways: to identify promising subsets of documents — to be perused with other tools or reclustered into more refined groups —, to identify exemplars for relevance feedback, or to eliminate groups of documents whose contents appear irrelevant.

In an informal exploratory paper [11] we outlined two examples of the application of Scatter/Gather to retrieval results. These anecdotes demonstrated that clusters are customized to the target collection. For example, Scatter/Gathering all encyclopedia articles containing the word *star* produced clusters with themes such as astronomy and astrophysics, animals and plants (star-shaped), and film stars. When the documents in the clusters about astrophysics and astronomy are gathered and then rescattered, the resulting clusters separate out the biographical articles from the rest, demonstrating a change of theme specific to that subcollection.

To examine these issues more closely, we constructed a graphical user interface that integrated Scatter/Gather, Tilebars [12], and similarity search, and used this interface in experiments for the TREC-4 interactive track [10]. We observed from subject interaction with this system that relevant documents tended to fall into one or two clusters (out of four possible offered by the interface), which helped subjects determine which subset of the collection to explore.

In the present paper we take this one step further and systematically evaluate Scatter/Gather as a method for viewing retrieval results. We compare the results of applying

[*] Authors listed in alphabetical order.

[1] The specific clustering algorithm employed may also affect the topics seen, however most clustering algorithms will yield roughly similar results if asked to produce the same number of groups. Scatter/Gather typically employs a fast, linear-time clustering algorithm. See [4, 5] for details

Scatter/Gather to that of similarity search ranking alone and find significant improvements using the clustering. In contrast to pervious work, this provides evidence validating the cluster hypothesis [23] which states that relevant documents tend to be more similar to each other than to non-relevant documents. Although we do not intend Scatter/Gather be used in isolation, but rather as one tool in a larger information workspace, we find it encouraging that clustering improves results when evaluated as a direct alternative to ranked titles.

In the following sections we discuss related work applying document clustering information retrieval, the architecture of our interactive system, an example of Scatter/Gather in use, and two evaluations. The first evaluation compares selecting a best cluster to an equivalent cutoff in ranked retrieval results. The second examines whether users select the best cluster.

2 Related Work

There has been extensive research on how to employ clustering to improve retrieval results. In most previous attempts the strategy was to build a static clustering of the entire collection and then match the query to the cluster centroids [26]. Often a hierarchical clustering was used and an incoming query was compared against each cluster in either a top-down or a bottom-up manner. (The relative success of the traversal direction seems to depend to some extent on the kind of clustering used [26].) The top-ranked clusters were chosen either by a cutoff in the number of clusters to be selected or by a cutoff in the query-centroid similarity score. Some variations on this theme were explored, for example, Croft [3] suggested a strategy in which a document that had a high similarity score with respect to the query would first be retrieved using a standard search and ranking method, and then this document used for the comparison to the cluster centroids.

Using clustering from this point of view, if a query did not match a cluster, that is, if a query was not a good representative of one of the pre-defined categories, then it would fail to match any of the existing clusters strongly. This problem was recognized and as a remedy Worona [27] suggested grouping previously encountered queries according to their similarity. A new incoming query that was not similar to any of the cluster centroids might instead be similar to one of the query groups, which in turn might be similar to a cluster centroid. (This strategy has been revisited recently in work on collection fusion by Voorhees et al. [25].)

In most experiments, retrieving the contents of the clusters whose centroids most closely match the query did not perform as well as retrieving the top ranked documents from the collection as a whole. Lower scores were reported in, e.g., [19, 8]. Croft [3] described a method of bottom-up cluster search which could be shown to perform better than a full ranking with cosine correlation on the Cranfield collection, provided that the cutoff used was one that had already been determined to be optimal for the clustering, and only when emphasizing precision over recall in the evaluation measure. Some of van Rijsbergen's experiments suggested that if the optimal cluster was chosen then results could be improved, although actual performance was inferior [13]. However, these studies were done on a subset of the Cranfield collection. In Willett's extensive survey article on document clustering [26], he finds problems with these and related studies, because they concentrated on the use of the (small) Cranfield collection and typically employed an evaluation measure in which only one small cluster would be chosen, typically selecting only two or three documents for retrieval. Willett summarizes the results of several different experiments he was involved with as showing that clustering does not outperform noncluster searches, except on the Cranfield collection. Voorhees [24] introduced a new way to evaluate whether or not the cluster hypothesis should hold, and tested it on several collections other than Cranfield, but was not able to find an improvement using clustering with this strategy.

In this paper we show for the first time that the application of clustering in Scatter/Gather can significantly improve retrieval results over a very large text collection. We conjecture that there are two main reasons for this deviation from previous work. The first is a familiar one: that the earlier work was done at a time when no large text collections with queries and relevance judgments was available. The fact that our experiments are run over a very large text collection [9], and also that the many of the documents are full text documents, as opposed to titles and abstracts, may also have some effect.

However, perhaps the more telling reason is that clustering is used in a different way in Scatter/Gather on retrieval results than in the earlier work.

Initially clustering was suggested both for reasons of efficiency – since matching against centroids might be more efficient than matching against the entire collection [26] – and as a way to categorize or classify documents. Salton and his coworkers did early experimentation with document clustering, viewing clustering as classification of documents in analogy to bibliographic subject headings. Salton wrote [19]:

> *In a traditional library environment, answers to information retrieval requests are not usually obtained by conducting a search throughout an entire document collection. Instead, the items are classified first into subject areas, and a search is restricted to items within a few chosen subject classes. The same device can also be used in a mechanized system by constructing groups of related documents and confining the search to certain groups only.*

Thus the classifications were intended to reflect an external reality about how to group the documents as well as what kinds of queries would be received, perhaps by the heavy reliance on subject codes in bibliographic search [15].

Perhaps as a consequence, clustering experiments have always assumed the clustering is done over the entire collection in advance, independent of the user's query. Van Rijsbergen explicitly voiced this assumption [23] (Ch. 3):

> *Another good example of the difference between experimental and operational implementations of a classification is the permanence of the cluster representatives. In experiments we often want to vary the cluster representatives at search time. ... Of course, were we to design an operational classification, the cluster representatives would be constructed once and for all at cluster time.*

He continued by emphasizing the importance of maintaining the same cluster structure as new documents are added to the collection.

By contrast, clustering in Scatter/Gather is dynamic, and the clusters that result are very much a consequence of which documents were retrieved in response to the query. As shown in the example in Section 3, different clusters arise given different result sets. (Scatter/Gather clustering is dynamic in another way as well; the user interacts with and manipulates the representation to understand something of the structure of the retrieval results.)

Thus our work supports the Cluster Hypothesis, but with some assumptions revised. We too assume that documents that cluster together are similar in some ways, and thus relevant documents will tend to cluster near other relevant documents and farther away from nonrelevant ones. However, in contrast with the assumption underlying the strategy of earlier work, we do *not* assume that if two documents D_1 and D_2 are both relevant or nonrelevant for query Q_A, they must also *both* be relevant or nonrelevant for query Q_B. That is, in our use of Scatter/Gather on retrieval results, the clusters are created as a function of which documents were retrieved in response to the query, and therefore have the potential to be more closely tailored to characteristics of a query than an independent, static clustering. In other words, because documents are very high-dimensional, the definition of nearest neighbors will change depending on which neighbors are on the street.

3 System Description

The system that is the subject of the experiments described in this paper consists of the Text DataBase (TDB) [6] engine developed at PARC and a graphical user interface that offers users a similarity search facility to resolve queries, and a choice of ranked titles, Scatter/Gather, or TileBars[12] to display results. TDB is implemented in Common LISP and CLOS, and the interface is implemented in TCL/TK [17]. The two parts communicate with one another through ILU [2] and expectk [16].

A flow diagram of the expected use of this system is shown in Figure 1. First the user specifies a query. A threshold n is set indicating the number of documents to be initially retrieved. The query is resolved as a similarity search and the top n documents returned, in rank order, which are then shown to the user in Title Mode.[2]

At this point, the user can switch the results display mode to be either, TileBars, and Scatter/Gather, or back to Titles. Since this paper focuses on the Scatter/Gather display method, we will not further discuss TileBars (see [11] for a discussion of the use of TileBars in this system).

The user can view a subset of the retrieval results by selecting one or more of the clusters produced by Scatter/Gather, indicating that only the contents of those clusters are to be viewed. The system maintains sufficient state so that it is possible for the user to back up, effectively undoing a cluster selection. The user may view the titles within one or more clusters by selecting them, restricting focus to that subset, and changing the display to Title or TileBar mode. The user is free to reformulate and reissue the query if desired. For experimental purposes the user is requested to mark documents as relevant, which marks that document in all displays. At the end of the session, the documents marked relevant are saved to a log file.

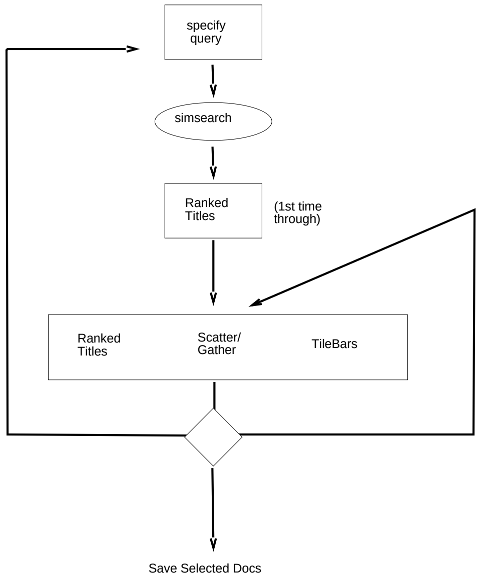


Figure 1: A flow diagram of the process model for the Interactive Track Interface.

3.1 The Document Clustering Algorithm

Scatter/Gather employs an automatic clustering algorithm to organize a set of documents into a given number of topic-coherent groups. The inputs into the clustering algorithm are a pairwise similarity measure (we use the cosine between document vectors) and the number of desired clusters, which, for these experiments, is set by the interface.

There are a wide variety of document clustering algorithms available (see, for example, [26]). We use a non-hierarchical partitioning strategy, called Fractionation [4], that clusters n documents into k groups in $O(kn)$ time. Fractionation is deterministic (i.e. the same groups are output given the same ordered document set), but order dependent (i.e. the groups do depend on the order of the document set).

Scatter/Gather is an interactive tool, hence Fractionation is optimized for speed rather than accuracy (in the sense of misclassification rate). The current system is capable of organizing 5000 short documents into five groups in under one minute elapsed time on a SPARC20 workstation.

3.2 A Worked Example

Suppose the user is interested in electric cars[3]. The first action is to issue a query, which is this case consists of the

[2] We use a simple cosine ranking scheme with tf.idf weights. In particular the similarity of document d to query q is computed as

$$S(d,q) = \frac{\sum d(w)q(w)}{\sqrt{\sum d(w)^2}}$$

where $d(w) = \sqrt{f_d(w)}$, $q(w) = \sqrt{f_q(w)}\ \log(N/n(w))$, $f_x(w)$ is the frequency of w in x and $n(w)$ is the number of documents in which w occurs

[3] This is reminiscent of TREC-4 topic 230, which reads "Is the automobile industry making an honest effort to develop and produce an electric-powered automobile?"

search terms *auto, car, vehicle* and *electric*[4], and feed it into similarity search which returns the 250 top-scoring documents from the TIPSTER [9] corpus in rank order . Initially these results are presented as ordered titles. However, the user may request them to be clustered in which case the display appears as in Figure 2.

Let us examine each of the five clusters in Figure 2. The first is quite small and consists of 8 documents having to do with safety and accidents, auto maker recalls, and a few very short articles. The second cluster is medium-sized at 25 documents and its central terms are related to alternative energy cars, including *battery, California* (the state that leads U.S. efforts towards alternative energy autos), *recharge, government,* and *cost.* The first five visible documents all appear to focus on the topic of electric cars. Note that *government* and *cost* terms are also important, because the government enforcement of alternative energy autos is one of the main drivers behind it, and cost of manufacturing such cars one of the main detractors.

The third cluster is the largest at 48 documents, and its focus can be seen to be on issues surrounding sales, economic indicators, and international trade, particularly issues surrounding imports by the U.S. and U.S. domestic sales. The fourth cluster is smaller at 16 documents and although also related to trade, focuses on exports from other countries. Thus an interesting distinction is made by the clustering between imports and exports. The last cluster acts as a "junk" group, holding three documents that are extremely short, and hence difficult to classify. Some of these are short article abstracts with no associated titles. It is our experience that small junk clusters of this sort occur fairly frequently and provide the useful service of separating out articles unlikely to be relevant due to their extreme brevity or due to their lack of similarity to the other clusters.

Given this display the user may make the decision to focus interest on the second cluster which contains most of the relevant documents, effectively dropping from consideration the junk cluster and the others containing documents on topics other than that intended by the query. Note, the user can revisit the discarded clusters later, if desired, by evoking the backup feature of the interface.

To highlight the shift in clustering possible given a slightly different document set consider Figure 3 which displays the results for the same query with *safety* substituted for *electric.* As in the results for *electric,* the search on *cars* and *safety* turns up a large cluster (Cluster 4) on imports and a small cluster (Cluster 5) on exports. Thus, the ranking algorithm is bringing up similar sets of documents that are not central to the intentions of the query, which are treated similarly in both cases. However, now instead of the one small cluster on accidents and the medium sized cluster on alternative energy vehicles, we see two clusters that focus on various aspects of auto safety. The first (Cluster 2) consists of articles about investigations into and hearings about various auto safety issues. The second (Cluster 3) contains articles discussing general issues surrounding auto safety, its connection to high gas-mileage cars (which are typically smaller and lighter, and therefore potentially more dangerous in an accident). Cluster 1 contains very short documents, similar to Cluster 5 in the previous example, as well as a few articles on auto accidents. If we had started with a static clustering we would not have been able to achieve this shift in groupings that are so tailored to the query.

4 Experiments

In this section we evaluate the ability of Scatter/Gather to successfully group together relevant documents, and separate them from nonrelevant ones. We do this by comparing a ranking of documents in a best cluster to an equivalent cutoff in the original ranked retrieval results. That is, we take the top n documents produced from similarity search, cluster them, score each cluster according to the number of relevant documents contained in the cluster, select the highest scoring cluster and compare the documents in that cluster (ranked two ways: by closeness to cluster centroid and by closeness to the query) to an equivalent number of documents from the original top n. This reflects the best-case scenario of a user viewing a set of clusters and selecting the most appropriate one for further examination. We discuss in the next section whether users can attain this best case performance given the cluster summaries generated by the system.

4.1 Collection

We experiment over the very large ($>$ 3 GB) TREC/Tipster standard reference collection [9] which consists largely of newswire, magazine articles, and government documents. Associated with this collection is a set of topic descriptions (referred to here interchangeably as queries) with matched relevance assessments. We use the TREC4 topics (202–250[5]) since they are short (a one sentence topic description) and hence better reflect an interactive ad hoc search situation. These are evaluated against disks 2 and 3 of the collection ($>$ 2 GB of text), since these cover the available relevance judgments. Given the size of this collection, we can consider our results to scale.

4.2 Results

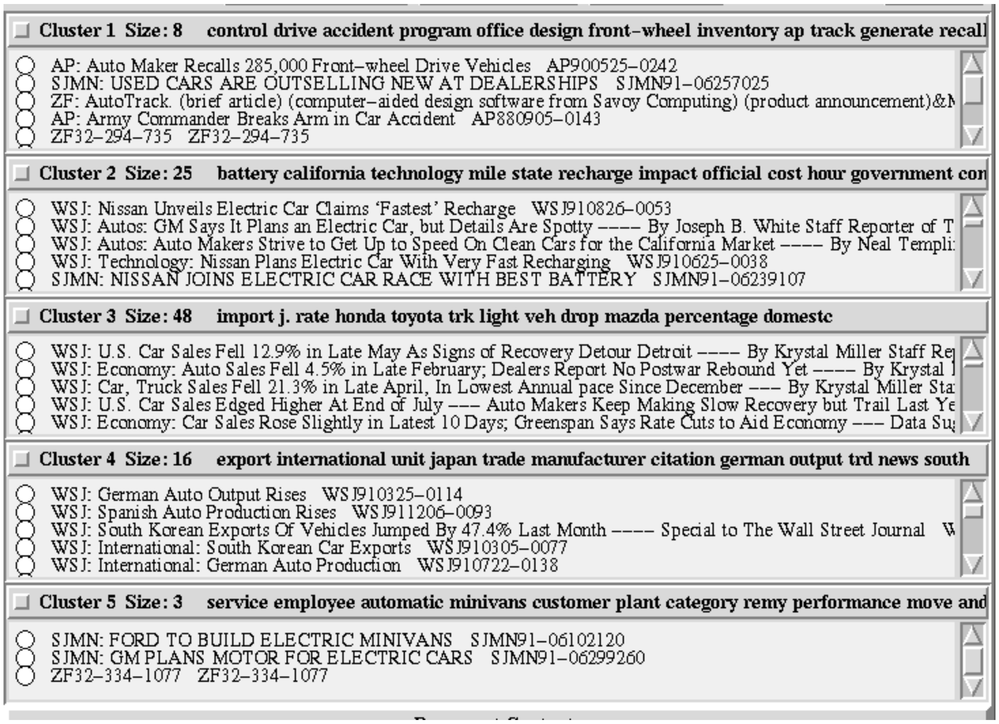
We took the 49 TREC-4 queries as originally written and retrieved the n top-ranked documents where n was set at 100, 250, 500, and 1000. These documents were then Scattered (or clustered) into 5 clusters. The number of clusters was chosen arbitrarily, but reflects users' preference for smaller numbers of clusters (see below).

Figure 4 displays the distribution of the percentage of relevant documents found for each ranked cluster. To obtain this figure the clusters resulting from each query are sorted according to the percentage of relevant documents they contain (in descending order). For example, the rank 1 cluster is the best cluster, that is, the one with the largest number of relevant documents. For each rank we display a boxplot of the distribution of the percentage of relevant documents found per query. From the strongly non-linear decrease in the medians of these distributions we see that the best clusters contain by far the largest proportion of relevant documents. In fact, the top-ranked cluster almost always contains at least 50% of the relevant documents retrieved, and usually a much larger percentage. The third, fourth, and fifth-ranked clusters usually contain 10% or fewer.[6]

[4] Actually, this is a conjunct of disjuncts when viewed in TileBar Mode, but is treated as a flat set of terms for query resolution purposes.

[5] Topic 201 has been thrown out by the TREC sponsors.

[6] See [18] for an earlier discussion of this behavior in a somewhat different setting.



□ **Cluster 1 Size: 8 control drive accident program office design front–wheel inventory ap track generate recall**

○ AP: Auto Maker Recalls 285,000 Front–wheel Drive Vehicles AP900525–0242

○ SJMN: USED CARS ARE OUTSELLING NEW AT DEALERSHIPS SJMN91–06257025

○ ZF: AutoTrack. (brief article) (computer–aided design software from Savoy Computing) (product announcement)&N

○ AP: Army Commander Breaks Arm in Car Accident AP880905–0143

○ ZF32–294–735 ZF32–294–735

□ **Cluster 2 Size: 25 battery california technology mile state recharge impact official cost hour government con**

○ WSJ: Nissan Unveils Electric Car Claims 'Fastest' Recharge WSJ910826–0053

○ WSJ: Autos: GM Says It Plans an Electric Car, but Details Are Spotty –––– By Joseph B. White Staff Reporter of T

○ WSJ: Autos: Auto Makers Strive to Get Up to Speed On Clean Cars for the California Market –––– By Neal Templi

○ WSJ: Technology: Nissan Plans Electric Car With Very Fast Recharging WSJ910625–0038

○ SJMN: NISSAN JOINS ELECTRIC CAR RACE WITH BEST BATTERY SJMN91–06239107

□ **Cluster 3 Size: 48 import j. rate honda toyota trk light veh drop mazda percentage domestc**

○ WSJ: U.S. Car Sales Fell 12.9% in Late May As Signs of Recovery Detour Detroit –––– By Krystal Miller Staff Re

○ WSJ: Economy: Auto Sales Fell 4.5% in Late February; Dealers Report No Postwar Rebound Yet –––– By Krystal

○ WSJ: Car, Truck Sales Fell 21.3% in Late April, In Lowest Annual pace Since December ––– By Krystal Miller Sta

○ WSJ: U.S. Car Sales Edged Higher At End of July ––– Auto Makers Keep Making Slow Recovery but Trail Last Ye

○ WSJ: Economy: Car Sales Rose Slightly in Latest 10 Days; Greenspan Says Rate Cuts to Aid Economy ––– Data Su

□ **Cluster 4 Size: 16 export international unit japan trade manufacturer citation german output trd news south**

○ WSJ: German Auto Output Rises WSJ910325–0114

○ WSJ: Spanish Auto Production Rises WSJ911206–0093

○ WSJ: South Korean Exports Of Vehicles Jumped By 47.4% Last Month –––– Special to The Wall Street Journal W

○ WSJ: International: South Korean Car Exports WSJ910305–0077

○ WSJ: International: German Auto Production WSJ910722–0138

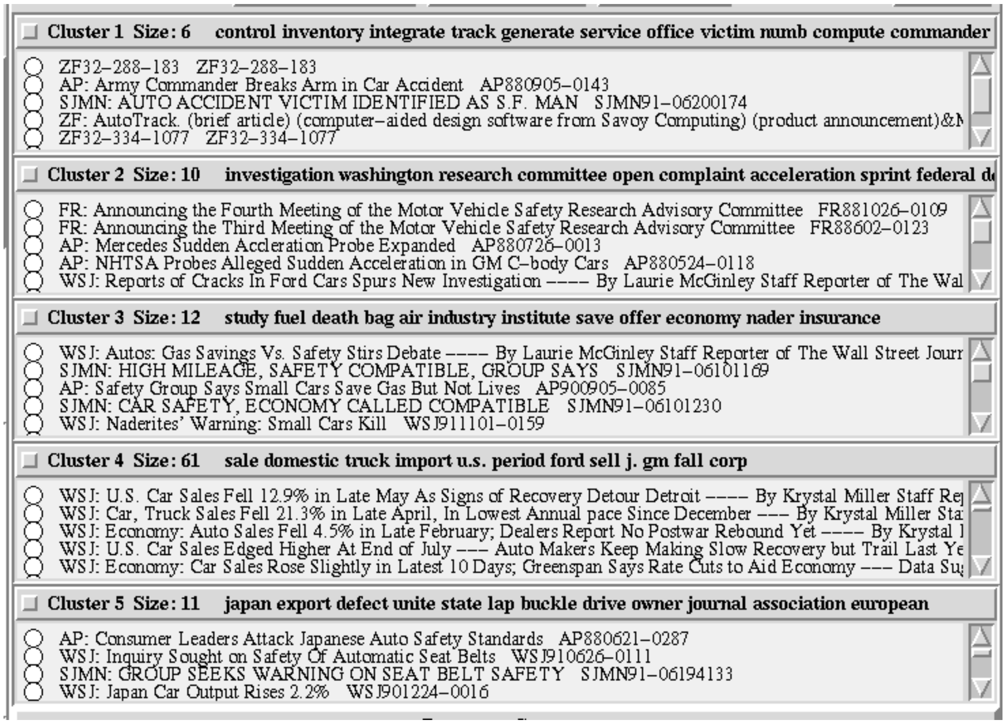
□ **Cluster 5 Size: 3 service employee automatic minivans customer plant category remy performance move and**

○ SJMN: FORD TO BUILD ELECTRIC MINIVANS SJMN91–06102120

○ SJMN: GM PLANS MOTOR FOR ELECTRIC CARS SJMN91–06299260

○ ZF32–334–1077 ZF32–334–1077

Figure 2: Scatter/Gather results on *auto, car, vehicle* and *electric* with a cutoff of 250.



□ **Cluster 1 Size: 6 control inventory integrate track generate service office victim numb compute commander**

○ ZF32–288–183 ZF32–288–183

○ AP: Army Commander Breaks Arm in Car Accident AP880905–0143

○ SJMN: AUTO ACCIDENT VICTIM IDENTIFIED AS S.F. MAN SJMN91–06200174

○ ZF: AutoTrack. (brief article) (computer–aided design software from Savoy Computing) (product announcement)&N

○ ZF32–334–1077 ZF32–334–1077

□ **Cluster 2 Size: 10 investigation washington research committee open complaint acceleration sprint federal de**

○ FR: Announcing the Fourth Meeting of the Motor Vehicle Safety Research Advisory Committee FR881026–0109

○ FR: Announcing the Third Meeting of the Motor Vehicle Safety Research Advisory Committee FR88602–0123

○ AP: Mercedes Sudden Accleration Probe Expanded AP880726–0013

○ AP: NHTSA Probes Alleged Sudden Acceleration in GM C–body Cars AP880524–0118

○ WSJ: Reports of Cracks In Ford Cars Spurs New Investigation –––– By Laurie McGinley Staff Reporter of The Wal

□ **Cluster 3 Size: 12 study fuel death bag air industry institute save offer economy nader insurance**

○ WSJ: Autos: Gas Savings Vs. Safety Stirs Debate –––– By Laurie McGinley Staff Reporter of The Wall Street Jourr

○ SJMN: HIGH MILEAGE, SAFETY COMPATIBLE, GROUP SAYS SJMN91–06101169

○ AP: Safety Group Says Small Cars Save Gas But Not Lives AP900905–0085

○ SJMN: CAR SAFETY, ECONOMY CALLED COMPATIBLE SJMN91–06101230

○ WSJ: Naderites' Warning: Small Cars Kill WSJ911101–0159

□ **Cluster 4 Size: 61 sale domestic truck import u.s. period ford sell j. gm fall corp**

○ WSJ: U.S. Car Sales Fell 12.9% in Late May As Signs of Recovery Detour Detroit –––– By Krystal Miller Staff Re

○ WSJ: Car, Truck Sales Fell 21.3% in Late April, In Lowest Annual pace Since December ––– By Krystal Miller Sta

○ WSJ: Economy: Auto Sales Fell 4.5% in Late February; Dealers Report No Postwar Rebound Yet –––– By Krystal

○ WSJ: U.S. Car Sales Edged Higher At End of July ––– Auto Makers Keep Making Slow Recovery but Trail Last Ye

○ WSJ: Economy: Car Sales Rose Slightly in Latest 10 Days; Greenspan Says Rate Cuts to Aid Economy ––– Data Su

□ **Cluster 5 Size: 11 japan export defect unite state lap buckle drive owner journal association european**

○ AP: Consumer Leaders Attack Japanese Auto Safety Standards AP880621–0287

○ WSJ: Inquiry Sought on Safety Of Automatic Seat Belts WSJ910626–0111

○ SJMN: GROUP SEEKS WARNING ON SEAT BELT SAFETY SJMN91–06194133

○ WSJ: Japan Car Output Rises 2.2% WSJ901224–0016

Figure 3: Scatter/Gather results on *auto, car, vehicle* and *safety* with a cutoff of 250.

bucket	mean	expected	t-value
1–20	.802	.320	11.2
21–40	.894	.269	8.68
41–120	.737	.242	13.38

Table 1: Comparison of observed percentage of relevant documents in best cluster against expected if relevant documents were distributed uniformly

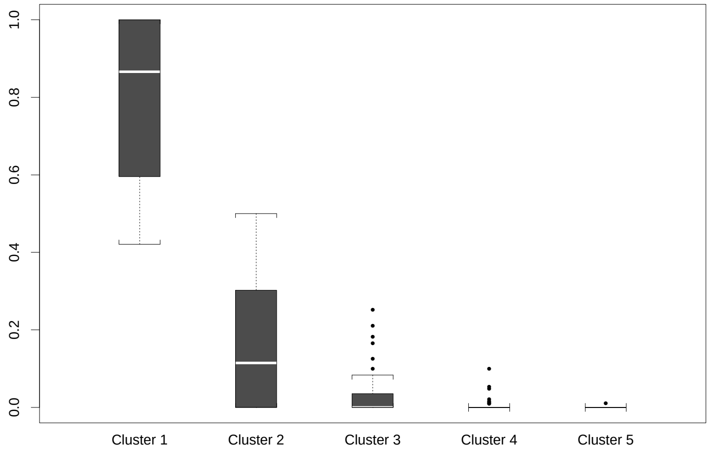


Figure 4:

Table 1 summarizes this information for cutoff 250. The 49 queries are placed into three buckets depending on the number of relevant documents found at that cutoff. The ranges are respectively 1 to 20 relevant documents (29 queries), 21 to 40 relevant documents (6 queries), and 41 to 120 relevant documents (14 queries). For each bucket we present the observed mean of the percentage of relevant documents found in the best cluster, the expected value of that number if relevant documents were distributed uniformly across clusters, and the t-value of the difference.[7] Every difference is significant at the .01 level which indicates that the distribution of relevant documents across clusters is far from uniform.

This suggests that clustering does in fact group together the relevant documents, as would follow from the Cluster Hypothesis.

If the best cluster is selected, we can compare it to the original ordering by truncating that ordering at the same number of documents and computing a measure of performance, in this case average uninterpolated precision, or the average precision at every relevant document (including those relevant documents not retrieved).[8] Table 2 presents the results for two different ordering of the documents in the best cluster: by closeness to the query (Cluster-Q) and by closeness to the cluster centroid (Cluster-C).

	Precision at Cutoffs		
CutOff	Sim-Ranked	Cluster-Ranked	% Increase
5	.342	.428	.252
10	.314	.401	.277
20	.276	.363	.312

Table 4: Precision at small document cutoff levels for the one-step algorithm.

The t-values are computed by variance normalizing the average per-query difference between methods (a paired t-test). T-values in excess of 2.40 are significant at the 2% level for a t-distribution with 48 degrees of freedom. This indicates that both methods that use clustering plus ranking significantly outperform similarity ranking alone for most values of n. The effect decreases with increasing n, until finally at $n = 1000$ the cluster centroid method is actually inferior to simple ranked titles. This can be explained by noting that as n increases the cluster sizes increase as well (since the number of clusters is fixed at 5). Large clusters are less able to sharply define a topic of interest especially if we rank within a cluster by closeness to the cluster centroid. It is generally the case that ranking within a cluster by nearness to a query performs better than ranking by nearness to the cluster centroid (taking into account the t-values).

For large values of n users typically take two Scatter/Gather steps. That is, they cluster the original n documents, select one or more clusters and recluster the indicated subset. This reduces the size of clusters to be examined by roughly a factor of five. We evaluate this strategy in much the same way as above. With this modification, the the best cluster is actually the result of two clustering steps (the best of the best). Results are shown in Table 3. Again both clustering plus ranking methods significantly outperform ranked titles, with ranking within clusters by nearness to query outperforming ranking by nearness to cluster centroid. It is interesting to note that the performance for this two-step procedure is similar to that of the one-step procedure if one divides n by five, which indicates that performance is strongly related to cluster size with 20 to 50 being close to optimal. (Other researchers have also found smaller clusters to yield better results than large ones [26].)

We also compared precision at small document cutoff levels for both the original ranked list and the best cluster. Table 4 presents the results averaged over all values of n (100, 250, 500, and 1000 documents, as before) for similarity search and the best cluster ranked by nearness to the query. Again, the clustering method consistently out performs ranked titles at all cutoff levels.

5 A User Study

For the TREC-4 interactive track [11] we presented participants (otherwise known as subjects) with the user interface

[7] The expected value is computed by considering the approximate distribution of the largest of five values given that the five values are generated by distributing the available relevant documents uniformly across the clusters.

[8] Average uninterpolated precision is a stringent measure at low cutoffs since all unretrieved relevant documents are averaged in at precision zero. Hence the small averages reported in Tables 2 and 3. Also note that the ranking scheme employed here does not use pseudo-feedback or other query expansion methods and hence is a relatively low baseline.

	Average uninterpolated precision						
n	Sim-Ranked	Cluster-Q	% Increase	t-value	Cluster-C	% Increase	t-value
100	.020	.031	.534	3.85	.036	.750	4.52
250	.027	.044	.637	4.42	.048	.791	4.14
500	.033	.045	.360	3.81	.047	.438	2.84
1000	.039	.046	.186	2.43	.038	-.015	-0.10

Table 2: Comparison of ranked titles to clustered and then ranked titles. Cluster-Q refers to documents within the best cluster ranked by similarity to the query. Cluster-C refers to ranking with respect to nearness to the cluster centroid.

	Average uninterpolated precision						
n	Sim-Ranked	Cluster-Q	% Increase	t-value	Cluster-C	% Increase	t-value
500	.023	.043	.831	4.05	.044	.862	3.39
1000	.029	.045	.556	3.93	.043	.504	2.57

Table 3: Comparison between ranked titles and two Scatter/Gather steps. Notations are as in Table 2.

described in Section 3. Here we report the results of analyzing how often, after issuing a search and clustering the results with Scatter/Gather, the users chose the cluster with the largest number of relevant documents to view next. We only looked at the choice made after the initial search and clustering, because after the participant has found many relevant documents, they may choose to explore clusters that are less likely to have relevant documents in an attempt to improve recall.

5.1 Experimental Setup

Our study consisted of four UC Berkeley graduate students, each of whom executed 13 queries. These consisted of 12 of the required 25, as well as one extra query given to all four participants. Only two of the participants' results on this query were reported, chosen arbitrarily. Participants completed queries in two sessions. The experiments were run in an otherwise empty room with a video camera recording the session. Participants were given an 10-minute demonstration of the interface followed by a 10-minute warmup exercise, and the participants were provided with a 3 page description of the interface for reference. Additionally, a binder of topic descriptions was prepared for each participant, with each topic description appearing on the top of a separate page. Participants were not allowed to look at a new topic before the current one was completed.

The instructions for the task were given as in the interactive track specifications: "find as many good documents as you can for a topic, in around 30 minutes, without collecting too much rubbish." We took this as a hard time limit; participants were required to stop when the 30 minute time limit was up. This statement emphasizes the finding of many relevant documents and deemphasizes the undesirability of including nonrelevant documents, and this had ramifications for how the participants performed. Some participants saved large numbers of documents for some of the queries without checking carefully for relevance, thus lowering overall precision.

5.2 Analysis of Participants' Use of Scatter/Gather

We found in 31 out of 38 cases, the participants chose the cluster with the largest number of relevant documents (if there was a tie for largest choosing any cluster with this number was counted as choosing the highest).[9] We have omitted from this calculation those cases in which no relevant documents are evident. In 6 of these cases the participants chose three clusters, in 10 they chose two clusters, and in the remaining 22 cases they selected only one cluster. In all seven cases in which the top-ranked cluster was not chosen, the participants chose only one cluster.

This is only an indirect reading of how informative the cluster summaries are, since the participants were not actually instructed to choose which cluster(s) they thought would have relevant documents, and a cluster might have been chosen because it looked interesting for some reason other than for the purposes of answering the query. Furthermore, because participants sometimes chose multiple clusters, we cannot assume that they would have chosen the best cluster if only one had been selected. Nevertheless, the large proportion of successful choices leads us to believe that users are able to take advantage of the benefits that clustering can provide.

5.3 Excerpts from Transcripts

After the sessions the participants were interviewed about the use of the interface, and the results of these interviews were recorded and transcribed.

When asked how and when they used the Scatter/Gather display, the participants said they mainly used them to narrow down the set of articles to be viewed with TileBars and to eliminate unpromising documents. Large clusters were often reclustered. None of the participants thought having more than five clusters would be a good idea. Some users said they interwove the use of Scatter/Gather and TileBars. One participant was especially enthusiastic toward the clustering, finding the clusters useful for weeding out nonrele-

[9] We determined which clusters contained the largest number of relevant documents by looking only at the first 11 documents in each cluster (or fewer if the cluster contained fewer documents), because the full cluster information was not recorded in our logs.

vant documents, but did express concern about tossing out appropriate documents.

6 Conclusions and Future Work

We have presented strong evidence that the Scatter/Gather approach to document clustering is one which can produce significant improvements over similarity search ranking alone. We have discussed the relationship of our approach to the use of clustering in previous work, and have concluded that, along with the use of a very large text collection, the most important difference is that our approach produces clusters that are tailored to characteristics of the query, rather than assuming that clusters play a one-size-fits-all, classificational role. Thus, this result provides evidence supporting the Cluster Hypothesis, that relevant documents tend to be more similar to each other than to non-relevant documents, if we add a new assumption: the same set of documents may behave differently in different contexts. We honor this assumption by performing clustering *after* the initial search and ranking.

We have also shown that users are able to successfully interact with the clustering produced by Scatter/Gather. They made extensive use of this mode of viewing retrieval results, and chose it over the option of using ranked titles alone. Furthermore, we have preliminary evidence that they were able to interpret the cluster summary information well enough to select the cluster with the largest number of relevant documents in most cases (although sometimes along with other clusters, since they were not asked specifically to select the best cluster).

In future we hope to perform more detailed user studies in order to determine in more detail how users make use of the Scatter/Gather representation.

References

[1] Matthew Chalmers and Paul Chitson. Bead: Exploration in information visualization. In *Proceedings of the 15th Annual International ACM/SIGIR Conference*, pages 330–337, Copenhagen, Denmark, 1992.

[2] A. Courtney, W. Janssen, D. Severson, M. Spreitzer, and F. Wymor. *Inter-Language Unification, release 1.5*. Xerox PARC, 1994. ftp://ftp.parc.xerox.com/pub/ilu/ilu.html.

[3] W.B. Croft. A model of cluster searching based on classification. *Information Systems*, 5:189–195, 1980.

[4] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proc. 15th Annual Int'l ACM SIGIR Conference on R&D in IR*, June 1992. Also available as Xerox PARC technical report SSL-92-02.

[5] Douglass R. Cutting, David Karger, and Jan Pedersen. Constant interaction-time Scatter/Gather browsing of very large document collections. In *Proceedings of the 16th Annual International ACM/SIGIR Conference*, pages 126–135, Pittsburgh, PA, 1993.

[6] Douglass R. Cutting, Jan O. Pedersen, and Per-Kristian Halvorsen. An object-oriented architecture for text retrieval. In *Conference Proceedings of RIAO'91, Intelligent Text and Image Handling, Barcelona, Spain*, pages 285–298, April 1991. Also available as Xerox PARC technical report SSL-90-83.

[7] Richard H. Fowler, Wendy A. L. Fowler, and Bradley A. Wilson. Integrating query, thesaurus, and documents through a common visual representation. In *Proceedings of the 14th Annual International ACM/SIGIR Conference*, pages 142–151, Chicago, 1991.

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

[9] Donna Harman, editor. *Proceedings of the Third Text Retrieval Conference TREC-3*. National Institute of Standards and Technology Special Publication 500-225, 1995.

[10] Marti Hearst, Jan Pedersen, Peter Pirolli, Hinrich Schüetze, Gregory Grefenstette, and David Hull. Four TREC-4 Tracks: the Xerox site report. In Donna Harman, editor, *Proceedings of the Fourth Text Retrieval Conference TREC-4*. National Institute of Standards and Technology Special Publication, 1996. (to appear).

[11] Marti A. Hearst, , David Karger, and Jan O. Pedersen. Scatter/gather as a tool for the navigation of retrieval results. In Robin Burke, editor, *Working Notes of the AAAI Fall Symposium on AI Applications in Knowledge Navigation and Retrieval*, Cambridge, MA, November 1995. AAAI.

[12] Marti A. Hearst. Tilebars: Visualization of term distribution information in full text information access. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, Denver, CO, May 1995. ACM.

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

[14] Robert R. Korfhage. To see or not to see – is that the query? In *Proceedings of the 14th Annual International ACM/SIGIR Conference*, pages 134–141, Chicago, 1991.

[15] Ray R. Larson. Experiments in automatic library of congress classification. *Journal of the American Society for Information Science*, 43(2):130–148, 1992.

[16] Don Libes. expect: Curing those uncontrollable fits of interaction. In *Proceedings of the Summer 1990 USENIX Conference*, Anaheim, CA, June 1990.

[17] John Ousterhout. An X11 toolkit based on the Tcl language. In *Proceedings of the Winter 1991 USENIX Conference*, pages 105–115, Dallas, TX, 1991.

[18] Peter Pirolli, Patricia Schank, Marti A. Hearst, and Christine Diehl. Scatter/gather browsing communicates the topic structure of a very large text collection. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, Vancouver, WA, May 1996. ACM.

[19] G. Salton. Cluster search strategies and the optimization of retrieval effectiveness. In G. Salton, editor, *The SMART Retrieval System*, pages 223–242. Prentice-Hall, Englewood Cliffs, N.J., 1971.

[20] Gerard Salton. *Automatic text processing: the transformation, analysis, and retrieval of information by computer.* Addison-Wesley, Reading, MA, 1989.

[21] Anselm Spoerri. InfoCrystal: A visual tool for information retrieval & management. In *Proceedings of Information Knowledge and Management '93*, Washington, D.C., Nov 1993.

[22] R. H. Thompson and B. W. Croft. Support for browsing in an intelligent text retrieval system. *International Journal of Man [sic] -Machine Studies*, 30(6):639–668, 1989.

[23] C.J. van Rijsbergen. *Information Retrieval.* Butterworths, London, second edition, 1979.

[24] Ellen M. Voorhees. The cluster hypothesis revisited. In *Proceedings of ACM/SIGIR*, pages 188–196, 1985.

[25] Ellen M. Voorhees, Narenda K. Gupta, and Ben Johnson-Laird. The collection fusion problem. In Donna Harman, editor, *Proceedings of the Third Text Retrieval Conference TREC-3*, pages 95–104. National Institute of Standards and Technology Special Publication 500-225, 1995.

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

[27] S. Worona. Query clustering in a large document space. In G. Salton, editor, *The SMART Retrieval System*, pages 298–310. Prentice-Hall, Englewood Cliffs, N.J., 1971.