Adaptive Clustering of Search Results

Xuehua Shen[1], ChengXiang Zhai[2], and Nicholas J. Belkin[3]

[1] Google Incorporation
[2] Department of Computer Science, University of Illinois at Urbana-Champaign
[3] School of Communication, Information and Library Studies, Rutgers University

Abstract. Clustering of search results has been shown to be advantageous over the simple list presentation of search results. However, in most clustering interfaces, the clusters are not adaptive to a user's interaction with the clustering results, and the important question "how to optimize the benefit of a clustering interface for a user" has not been well addressed in the previous work. In this paper, we study how to exploit a user's clickthrough information to adaptively reorganize the clustering results and help a user find the relevant information more quickly. We propose four strategies for adapting clustering results based on user actions. We propose a general method to simulate different kinds of users and linearize the cluster results so that we can compute regular retrieval measures. The simulation experiments show that the adaptation strategies have different performance for different types of users; in particular, they are effective for "smart users" who can correctly recognize the best clusters, but not effective for "dummy users" who follow system's ranking of results. We further conduct a user study on one of the four adaptive clustering strategies to see if an adaptive clustering system using such a strategy can bring users better search experience than a static clustering system. The results show that there is generally no significant difference between the two systems from a user's perspective.

1 Introduction

The main goal of a search engine is to rank relevant documents above non-relevant ones. There has been a lot of research in developing effective retrieval models to help achieve this goal. However, an equally important goal of a search engine is to present the search results effectively so that a user can find the relevant information from the results quickly. Indeed, most search engines present a ranked list of documents with brief summaries. However, due to the inevitable mismatches between a query and documents, the search results are more often non-optimal. It is quite common that a user may not find any relevant document among the top ranked ones. In such a case, the user would have to go through the many non-relevant documents in the list until eventually finding some relevant ones. Intuitively, a clustering view of the search results would be much more useful in such a case.

Clustering of search results has been shown to be an effective way to present the search results [7, 2], and has been adopted by some search engines such as vivisimo [4]. Although clustering of search results has been studied, in most existing work, the clusters are generally not adaptive to a user's interaction with the clustering results. In Scatter/Gather[1, 3], the authors proposed that re-clustering can be performed on the user-selected clusters, which can be regarded as attempting to adapt the clustering results to

[4] http://vivisimo.com/

a user. Unfortunately, this is as far as the work goes, and to the best of our knowledge, there has been no work that attempts to seriously study the important question "how to optimize the benefit of a clustering interface for a user." With non-adaptive clustered results, a user would generally select a cluster and examine the results in it. Thus the utility of the results is largely determined by the clustering algorithm. Intuitively, however, the clustering results may be improved as the system sees more user interactions when the user examines the results.

In this paper, we study how to exploit a user's clickthrough information, which is naturally available when a user is interacting with a clustering interface, to adaptively reorganize the clustering results and help a user find the relevant information more quickly. Specifically, we propose four strategies for adapting clustering results based on user actions, which are (1) reranking documents based on a selected cluster, (2) reranking documents based on a viewed document, (3) merging unselected clusters, and (4) promoting "near-miss" documents. Evaluation of the utility of a cluster presentation of results is a challenging task. We propose a general method to simulate different kinds of users and linearize the cluster results so that we can compute regular retrieval measures. The simulation experiments show that the adaptation strategies have different performance for different types of users; in particular, they are effective for "smart users" who can correctly recognize the best clusters, but not effective for "dummy users" who simply follow system's ranking of results. Among the four proposed adaptation strategies, the strategy of reranking based on viewed document is shown to be most effective, but other strategies are also beneficial. We further conduct a user study to see if an adaptive clustering system can bring improved search experience to users, compared with a static clustering system. We focus on the strategy of promoting near-miss documents. The results show that there is generally no significant difference between the two systems from a user's perspective. Specifically, more users say that they like the adaptive system better but users saved more relevant documents with the static interface than with the adaptive interface. Overall, our study shows that adaptive clustering has a good potential for improving search utility for users, but a user may not perceive any significant difference in the system.

2 Four Adaptive Strategies

We propose four strategies for adapting clustering results based on user actions, including reranking documents based on a selected cluster, reranking documents based on a viewed document, merging unselected clusters, and promoting "near-miss" documents.

The first is reranking based on cluster selection. When a user selects a cluster to view, we may infer that the user likes the selected cluster better than un-selected cluster(s). This information is exploited to improve the ranking of documents within a cluster. The information about the selected cluster can be combined with the original query to rerank documents in other (unselected) clusters.

The second is reranking based on document selection. When a user clicks on a document to view after selecting a cluster, the viewed document can presumably provide more information about what the user is interested in to the retrieval system and can thus be exploited to improve search results. The query will be updated with selected document and used to rerank the documents within each cluster.

The third is merging unselected clusters. When the user selects a cluster or views a document, the adaptive clustering algorithm can also restructure the clusters. In [1, 3], the retrieval system merges several relevant clusters according to a user's selection. After seeing a user selecting a cluster, it would be reasonable to assume that the user may not be so interested in the partitioning of search results in other clusters. Thus, the retrieval system can merge all unselected clusters into a big cluster and put this big cluster below the selected cluster. Then according to the updated query, the retrieval system can rerank all the documents in the big cluster.

The fourth is promoting “near miss” documents. In this strategy, when the user clicks on a cluster, the adaptive clustering algorithm presents not only the documents in the clicked cluster to the user, but also some (borderline) documents which were originally scattered into unselected clusters. Specifically, the retrieval system would select those documents from each unselected cluster that are most similar to the updated query vector, and then insert these “near miss” documents into the bottom of the selected cluster.

The first and second strategies do not change the cluster structure while the third and fourth do.

3 Simulation Study

Following the use of the simulation strategy in some previous work [3, 6], we also use the simulation strategy to evaluate the proposed adapatation strategies. But different from previous work, our simulation distinguishes two different types of extreme users. One is “smart users”, who can always make intelligent decisions. Such a user would be assumed to always select the best cluster among several clusters according to the description of the cluster labels and always select a relevant document among a set of documents to view according to the snippet of each document. We simulate the interaction of the smart user as follows. We compute the percentage of relevant documents in each cluster, sort clusters according to the percentage of relevant documents in a cluster. We assume the smart user will select the top (also the best) cluster. We linearize or expand these clusters into a ranked list and evaluate the expanded ranked list with regular retrieval performance measures. We refer to the retrieval performance of this ranked list as the *smart baseline*. After a smart user selects the cluster with the highest percentage of relevant documents, the retrieval system can update the user’s information need and rerank documents in each cluster. We evaluate the performance of the new ranked list and refer to the result as the *smart adaptive*. After the documents of the best cluster are presented, the smart user would select the best document in the best cluster to view. We simulate this behavior by selecting the top ranked *relevant* document in the best cluster. Immediately after the smart user selects the best document in the best cluster to view, the retrieval system will further update the user’s information need and rerank documents in each cluster again. We refer to this result as the *smart reranking*. The other type of extreme users is “dummy users.” A dummy user would always select top ranked cluster and top ranked document to view. Such a user would just passively follow a system’s ranked result. In real web search, it is found that a user’s behavior has a view bias and clickthrough can be biased according to the presented ranking order by a search engine. The user tends to view or click on documents from the top. As in the case of a smart user, we can also define *dummy baseline*, *dummy adaptive*, and *dummy reranking* similarly. In the real world, a user can be considered as a mixture of smart user and dummy user. We can use the probability of being smart user, which varies from 0.0 (dummy user) to 1.0 (smart user) to control the user interaction behavior.

We use TREC8 ad hoc track data set for empirical evaluation. For each query, we use vector space model to obtain baseline retrieval results. We then use K-Medoids clustering algorithm to cluster the top 100 documents into 6 clusters. We employ a *linearization* method to “convert” any clustering results into a *perceived* ranked list. In each cluster, the documents will be sorted by the retrieval scores. The idea is to simulate the perceived order of documents by a user when the user is browsing a clustering result. This way, we can evaluate a clustering result in the same way as evaluating a regular ranked list of result. We use the mean average precision (MAP), precision at 0.1 (pr@0.1) and 0.2 (pr@0.2) recall levels, and precision at top 10 (pr@10d) and 20 (pr@20d) documents as the evaluation metrics.

Row 3-8 of Table 1 shows the retrieval performance of the experiment results for two types of extreme users at different stages. We can see that for the smart user, the adaptive clustering strategy is effective and the performance of smart baseline is much better than that of baseline. For the smart user, smart reranking is apparently better than smart adaptive and smart baseline while smart adaptive has a better pr@10d than smart baseline does. For the dummy user, however, the adaptive clustering strategy appears to be ineffective. The dummy baseline is not as good as the baseline, which means clustering presentation is not effective for dummy users. Dummy adaptive and dummy reranking have similar performances to dummy baseline. Thus the adaptive clustering representation is generally not effective for the dummy user. In the cluster regrouping

Table 1. Experiment Results of Simulation Study.

Method	MAP	pr@0.1	pr@0.2	pr@10d	pr@20d
baseline	0.230	0.459	0.355	0.398	0.356
smart baseline	0.283	0.529	0.413	0.531	0.465
smart adaptive	0.282	0.559	0.418	0.539	0.470
smart reranking	0.294	0.580	0.428	0.551	0.467
dummy baseline	0.205	0.410	0.324	0.357	0.318
dummy adaptive	0.196	0.414	0.324	0.349	0.326
dummy reranking	0.202	0.418	0.331	0.353	0.320
regroup adaptive	0.261	0.557	0.401	0.533	0.45
regroup reranking	0.270	0.578	0.407	0.539	0.453
promotion adaptive (q)	0.282	0.561	0.421	0.537	0.457
promotion reranking (q)	0.293	0.583	0.434	0.545	0.460
promotion adaptive ($q^{'}$)	0.287	0.575	0.428	0.527	0.471
promotion reranking($q^{'}$)	0.291	0.582	0.428	0.549	0.452

strategy, when the smart user selects one cluster, we will merge other clusters into one cluster so that there will be only two clusters – the active cluster and the unopened cluster. Immediately after the smart user selects a cluster to view, the retrieval system will update the user's information need and use the updated query to rerank the documents in each cluster. We call this result the *regroup adaptive*. When the user selects the best document to view, we can then rerank the documents in each cluster; the corresponding result will be called *regroup reranking*. Row 9-10 of Table 1 shows the retrieval performance of regroup adaptive and regroup reranking. We find that the retrieval performance of regroup adaptive and regroup reranking is not as good as the corresponding smart adaptive and smart reranking.

When we apply the promotion strategy to select a subset of documents from clusters other than the best cluster, we can use the original query to rank and select documents. We can also use the updated query, which is interpolated with the best cluster term vector, to rank and select documents. We tried both queries to promote documents. We promote one document from each cluster other than the best cluster and append it to the bottom of the best cluster. Row 11-14 of Table 1 shows the experiment results using this near miss promotion strategy. We can see that promotion reranking consistently has better retrieval performance than the promotion adaptive strategy. The better performance of promotion reranking over promotion adaptive clearly comes from reranking documents based on the viewed document; this observation is consistent with what we observed in the performance comparison of smart adaptive and smart reranking.

4 User Study

We further conduct a user study by deploying two clustering systems to real users and study whether adaptive clustering strategy can bring better user experience in interacting with the search results than the static clustering strategy. Among the four adaptive clustering strategies, the strategy of promoting "near-miss" strategy has shown some promising results. Thus we use this strategy in the adaptive clustering system.

We implement the clustering result presentation functionality in the UCAIR toolbar [4], which is an Internet Explorer plugin. The adaptive clustering strategy is also implemented in the UCAIR toolbar. Thus we evaluate two systems with clustering result presentation, i.e. Adaptive System (AS) and Static System (SS). We randomly select 6 query topics from TREC8 ad-hoc track topics. After the subjects submit title query to UCAIR toolbar, the UCAIR toolbar will return clustered results to the user by clustering top ranked 100 documents from Google into 6 clusters. We use the centroid document to represent each cluster. Subjects browse clusters and click one cluster to view snippets of documents which belong to the selected cluster. Subjects browse document snippets and then click the most interesting one to view the content of the document. If it is relevant, subjects will save it on the local disk. After a pilot study, 24 subjects partcipated in the formal study.

We collect the exit questionnaires and compare two systems by overall experiences including difference of two systems, helpfulness of systems in completing tasks, easiness of learning to use, easiness of using, and overall preference. The results are listed as Table 2. From Table 2, we can see that System AS is a little better than SS in the

Table 2. Comparison of Overall Experience of Two Clustering systems

Comparison	AS is better	SS is better	No difference
Helpfulness	10	6	8
Easy to Learn	1	3	20
Easy to Use	7	2	15
Overall Preference	11	5	8

aspect of being helpful in completing tasks and easy to use. Both systems are equally easy to learn since their interface is nearly identical. For the overall preference, System AS seems to be better than System SS. However, there is no clear indication that System AS is better than System SS. For more detailed analysis of simulation study and user study, please refer to [5].

5 Conclusion and Future Work

In this work, we explore adaptive clustering presentation in interactive information retrieval and propose four adaptation strategies to improve clustering results based on a user's implicit feedback information. We propose a stochastic way to simulate a user's browsing behavior and propose a method for evaluating clustering results quantitatively based on user simulation. We evaluate the proposed adaptation algorithms with simulation experiments and show that adaptive clustering, especially reranking of documents based on viewed document is effective for smart users who would intelligently identify and view a high precision cluster and pick a relevant document to view, though such strategies are not effective for dummy users who simply follow a system's ranking of clusters and documents. We further conduct a user study to see if an adaptive clustering system can bring improved search utility and/or experience to users, compared with a static clustering system. The results show that there is generally no significant difference between the two systems from a user's perspective. Specifically, more users say that they like the adaptive system better but users saved more relevant documents with the static interface than with the adaptive interface. Overall, our study shows that adaptive clustering has a good potential for improving search utility for users, but a user may not perceive any significant difference in the system.

There are two particularly interesting directions to explore. One direction is to look into other factors to affect the user experience. Although the user study comparing one method of adaptive clustering to static clustering showed no significant user performance or preference differences, this could be due to a variety of factors which we were unable to investigate in this study. Such factors include suitability of the clustering technique to the specific retrieval task; combining user behavior evidence for ranking as well as clustering; and combining several adaptation strategies, rather than using only one. The other one is to do a larger-scale of user study with more topics and multiple iterations of interaction to draw more reliable conclusions.

References

1. D. R. Cutting, J. O. Pedersen, D. Karger, and J. W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of SIGIR 1992.*, pages 318–329, 1992.
2. S. T. Dumais, E. Cutrell, and H. Chen. Optimizing search by showing results in context. In *Proceedings of CHI 2001*, pages 277–284, 2001.
3. M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: Scatter/gather on retrieval results. In *Proceedings of SIGIR 1996*, pages 76–84, 1996.
4. X. Shen, B. Tan, and C. Zhai. Implicit user modeling for personalized search. In *Proceedings of CIKM 2005*, pages 824–831, 2005.
5. X. Shen, C. Zhai, and N. J. Belkin. Adaptive clustering of search results in interactive information retrieval. Technical report, 2009.
6. R. W. White, I. Ruthven, J. M. Jose, and C. J. van Rijsbergen. Evaluating implicit feedback models using searcher simulations. *ACM Transaction of Information System*, 23(3):325–361, 2005.
7. O. Zamir and O. Etzioni. Grouper: A dynamic clustering interface to web search results. In *Proceeding of WWW 1999*, 1999.