# Incremental Model for Query Clustering

Poonam Goyal
Department of Computer Science
BITS Pilani, India 333 031
poonam@bits-pilani.ac.in

Mehala N
Department of Computer Science
BITS Pilani, India 333 031
mehala@bits-pilani.ac.in

Dipen Thakkar
Department of Computer Science
BITS Pilani, India 333 031
h2011154@bits-pilani.ac.in

## ABSTRACT
The traditional query clustering algorithms are designed to work on previously collected data for query stream. These algorithms become less and less effective with time because users interests, query meaning and popularity of the topics etc change over time. So there is a need for incremental model which can accommodate the concept drift that crop up with the new data being added to the collection without performing complete re-clustering. We have proposed an incremental model that periodically updates new information efficiently and can be applied to a distributed environment. Experiments, on TREC data set show that the model achieves accuracy very close to the static model.

## Categories and Subject Descriptors
H.3.3 [**INFORMATION STORAGE AND RETRIEVAL**]: Information Search and Retrieval - *Clustering, Information filtering, query formulation, Search process*

## General Terms
Algorithm, Performance and Experimentation.

## Keywords
Query recommendation, Search engine, Concepts, Click-through log, Query clustering, Document clustering.

## 1. INTRODUCTION
Query recommendation is a powerful tool to improve the search engine performance by allowing users to formulate queries in a more effective manner or to choose a query from the suggestions according to their intention. It is evident from the literature that the effective query clustering is the most powerful way for query recommendation. Query clustering groups semantically related queries and can be done by measuring the keywords similarity or by using the click-through logs, or both. Both the approaches have limitations as click-through log may have noisy information and users' needs may not be fully captured by query text and the relevant documents. Advantages of both the approaches are incorporated by considering both the factors in few algorithms [2,3]. Most of the feedback-content based clustering algorithms are designed to work on previously collected data for query stream. These algorithms become less and less effective with time because users interests, query meaning and popularity of the topics etc change over time. So there is a need for incremental algorithms which can accommodate the concept drift that crop up with the new data being added to the collection without performing complete re-clustering. To the best of our knowledge only few incremental models have been proposed for query clustering in the literature. The most of the existing algorithms poses one or more problems e.g. predefined number of clusters, clustering sensitive to the data order, session segmentation problem and parameter tuning etc. Hierarchical clustering algorithms have an additional advantage that these provide a view of the data at different levels of abstraction, which helps for people to visualize and interactively explore large document collections. Therefore, it would be better to consider and extend existing hierarchical algorithms into incremental hierarchical clustering algorithms, which can update new information effectively to leverage hierarchical nature of web data. In this paper, we have proposed an incremental model which can be applied to these algorithms.


*FIRE'12*, December 17–19, 2012, Kolkata, India.


Query-Document (QD) [1], Query-Concept (QC) [2] and Query-Document-Concept (QDC) [3] are some popular feedback-content based hierarchical clustering algorithms. In QD algorithm, first query-document bipartite graph is constructed, in which the vertices on one side are unique queries whereas on the other side are unique documents. A document which is clicked for a query is linked to it. An agglomerative clustering algorithm is applied to obtain clusters of similar queries and similar documents. In the QC algorithm, the bipartite graph has concepts nodes on the other side of the graph in place of documents in QD. Concepts in QC are features appearing in the web-snippets of the search results of queries. In QDC, first query-document-concept tripartite graph is constructed, where left and right side vertices are unique queries and concepts respectively and vertices between them are clicked documents. This graph structure is capable of decoupling the relation between queries and the concepts. It precisely stores information about all documents clicked for a query, all concepts extracted from a document and documents involved in sharing the concepts for different queries and the number of documents clicked for a query etc. An agglomerative clustering algorithm is applied by proposing three new measures to obtain clusters of similar queries and the respective set of similar documents and the respective sets of concept clusters. QDC clustering algorithm not only gives query clusters but also gives concept based good quality document clusters and concept clusters with respect to the formed query clusters. We have applied our incremental model to these algorithms. In the further sections, we have described about our proposed incremental model and experimental results.

## 2. INCREMENTAL MODEL
The proposed incremental model, which is shown in Figure1, can be described by the following steps: 1. Clustering algorithm is applied on the data collected at time $t_i$. Set of clusters are obtained as per the algorithm. 2. New batch of data is received at time $t_{i+1}$ and is clustered separately by the algorithm. 3. The two

sets of clusters from step1 and step2 are concatenated/merged and then clustered by the same algorithm. Resultant set of clusters are obtained at time $t_{i+1}$. Now the model is ready to receive next batch of data.

The distributed approach of the proposed model can cluster data on different nodes individually and can generate local models. Local models then can be merged to get the global set of clusters. The distributed approach will lead to the following advantages:

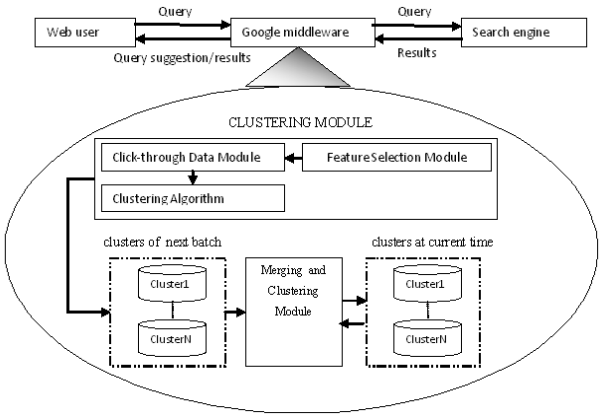


**Figure 1.** The proposed Incremental Model

only processed information need to be passed, privacy can be preserved, local completeness- clusters can be formed from the information scattered on local nodes.

The proposed approach is complete as it provides result as accurate as the result of clustering on the entire data set (including data at $t_i$, $t_{i+1}$,..). It also reduces the processing time many folds because clustering time for individual batch is less and merging and clustering two sets of clusters is very efficient in comparison to time taken by the clustering over the whole data.

## 3. EXPERIMENTAL RESULTS

We have implemented Google middleware for an evaluation of the proposed model. A query is submitted to the Google search engine through middleware without any modification and search results will be processed by the middleware and presented to the user. We have used TREC 2011 session track data set (http://trec.nist.gov/data/session2011.html) for the evaluation of our model. The TREC 2011 Session Track released 76 query sessions for 61 topics (some topics had more than one session corresponding to them). Each session consists of current query $q_m$ and a query session prior to the current query. Prior session has information about the set of past queries with their ranked URLs and clicked URLs. The data is partitioned into three partitions ($P_1$, $P_2$ and $P_3$) randomly for the incremental model evaluation. The measures Recall(R), Precision (P) and F-measure (F) are used for the comparison. These measures are calculated for the query clusters obtained from various algorithms and compared with our predefined clusters. The predefined clusters are manually formed clusters by placing the relevant queries in a cluster. We have applied the proposed incremental model on three algorithms QD, QC and QDC. The precision, recall and F-measures of QC and QDC algorithms are compared in Table 1 for static (namely QCS and QDCS respectively) and incremental algorithm (namely QCI and QDCI respectively). The incremental model is applied two times for addition of $P_2$ and $P_3$. It is observed that the incremental model gives the values of the measures very close to the values of static ones. Moreover, QDC clustering results are better than the QC results. We have not included the QD results as the TREC data set does not have enough common clicked documents. Therefore, it was not producing clusters. The time taken by the incremental model and static model is compared in Figure2 for QDC algorithm. It can be seen that time taken by the incremental model is much less than the static one. The incremental model time includes all individual clustering and merging. It is also observed that the incremental model for QDC algorithm not only gives good quality query clusters but also gives pure concept and document clusters.

**Table 1.** Best Precision, Recall and F-Measure at different time

| | $P_1$-$P_2$ | | | $P_1$-$P_2$-$P_3$ | | |
|---|---|---|---|---|---|---|
| | P | R | F | P | R | F |
| **QCS** | 0.556 | 0.611 | 0.582 | 0.842 | 0.506 | 0.632 |
| **QCI** | 0.695 | 0.5 | 0.582 | 0.889 | 0.456 | 0.603 |
| **QDCS** | 0.913 | 0.63 | 0.746 | 0.917 | 0.693 | 0.789 |
| **QDCI** | 0.923 | 0.596 | 0.724 | 0.951 | 0.665 | 0.783 |

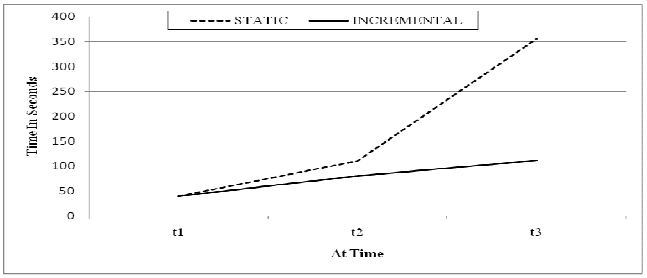


**Figure 2.** Time comparison for static and incremental model for QDC.

## 4. CONCLUSIONS

In this paper, we have proposed an incremental model for feedback-content based hierarchical algorithms for query clustering. The model achieves as high accuracy as the static model. The time taken by the incremental approach is much lesser than the static. It is found that the best results are obtained for the query-document-concept algorithm among the three algorithms considered in the experiments. The proposed model is able to assist web users to build a proper search query with the knowledge domain terminology to get the desired result.

## 5. REFERENCES


[1] Beeferman, D. and Berger, A. 2000. Agglomerative clustering of a search engine query log. In Proceedings of the sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Boston, MA USA, 2000) 407–416.

[2] Leung, K.W.-T., Ng, W. and Lee, D.L. 2008. Personalized Concept-Based Clustering of Search Engine Queries. IEEE Transactions Knowledge and Data Engineering. 2008; 20(11), pp. 1505-1518.

[3] Poonam Goyal and Mehala, N. 2012. A Robust Approach for Finding Conceptually Related Queries Using Feature Selection & Tripartite Graph Structure. Journal of Information Science, SAGE.