Enhancing Cluster Quality by Using User Browsing Time

Rehab Duwairi
Dept. of Computer Information Systems
Jordan Univ. of Sc. and Technology
Irbid, Jordan
rehab@just.edu.jo

Khaleifah Al.jada'
Dept. of Computer Science
Jordan Univ. of Sc. and Tech.
Irbid, Jordan.
khaleefh@hotmail.com

Abstract

The World Wide Web currently contains billions of documents; this causes difficulty in finding the desired information by users. Many search engines come out to help users finding their desired information but search engines still return hundreds of irrelevant web pages that do not fulfill the user's query. Several search engines use clustering to group documents that are relevant to the user's query before returning them to the user, but there is no document clustering algorithm that has an accuracy that can prevent retrieving irrelevant documents. In this research, the researchers have introduced a new technique to enhance cluster quality by using user browsing time as an implicit measure of user feedback, rather than using explicit user feedback as in previous research and techniques. The major contributions of this work are: investigating user browsing time as an implicit measure of user feedback and proving its efficiency, enhancing cluster quality by using a new clustering technique that is based on user browsing time, and developing a system that tests the validity of the proposed technique.

Keywords: Web Mining, Data Mining, Implicit Feedback, Clustering, Filtering, Search Engine.

1. Introduction

The Word Wide Web is the largest known collection of hypertext data, it is dynamic in nature and grows exponentially with time; for example, the number of documents published on the web reached 550 billion documents in 2001 [4]. This vast quantity of data makes finding proper information a tedious and time consuming activity for web users.

Most web users are using search engines to find their desired information on the web, however, search engines usually retrieve large number of web pages as a result of a user query; most of these pages are irrelevant to the user. Typically users browse through the first two pages of the returned list of pages. Search engines employ ranking algorithms that aim to placing the most relevant pages to the user's query into the top of the list of returned pages.

Many recommender systems (i.e. systems that aim to enhance search results to satisfy user's queries) which are based on user feedback were developed to solve the aforementioned problem, but these techniques use explicit user feedback which is costly in terms of time and resources. An implicit user feedback can be driven by using different features like click-through, time spent on the page (browsing time), exit type, added to favorites, and scrolling count [7]. Clustering can be employed by search engines to group together similar pages; subsequently these clusters can be utilized in ranking pages or in improving the presentation styles of search engines. In this research we investigate user browsing time as an implicit user feedback measure to enhance web search accuracy by enhancing the quality of web pages' clusters. The objective is to enhance web search precision based on user browsing time. The experiment shows an incentive enhancement on search precision after applying the proposed technique.

2. Background and Related Work

Explicit user feedback does not apply easily in different real-world information and filtering applications [7]. Recently many researchers are discussing and proposing techniques to infer user feedback implicitly. A study done by Steve Fox and others [7] shows that implicit user feedback can substitute for explicit user feedback. This study proposes different measures that can be used to infer user feedback**.** This research uses two of these measures namely: duration in seconds, and visits (number of visitors) to calculate average browsing time to implicitly infer user feedback**.**

Another technique is proposed in [6] where documents are represented in different ways (such as top ranked sentences (TRS), document title, summary sentence, sentence in context, and full text document). The idea of this technique is that the user starts by browsing TRS of the document if s/he is interested in it, s/he will browse the next representation which is the page title, and if s/he is still interested, s/he goes to the next representation until s/he reaches the document. The relevance degree is determined by observing the browsing path. As the user goes deeply in this path, the more relevant this document is.

Semi-supervised clustering allows users to provide their feedback [2]. The technique uses initially an unsupervised clustering algorithm to cluster documents, and then the user browses the resulting clusters and provides feedback to the system by saying:

- "This document doesn't belong in here".
- "Move this document to that cluster".
- "These two documents shouldn't be (or should be) in the same cluster".

To illustrate semi-supervised clustering, assume that two documents x_1 and x_2 were clustered into the same cluster by the employed algorithm, when the user browses these documents the user says "these two documents shouldn't be in the same cluster", consequently the clustering algorithm must modify the distance measure in the next iteration to increase the distance between x_1 and x_2 to separate them. Figure 1 illustrates how semi-supervised clustering works.

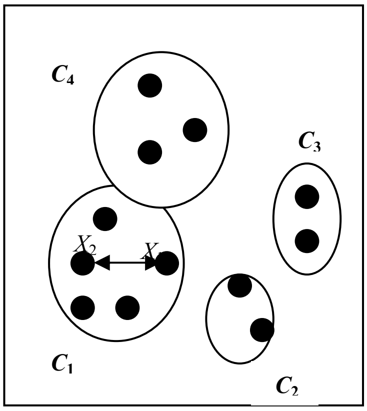

Figure 1 (a): Semi-supervised clustering *before* user feedback

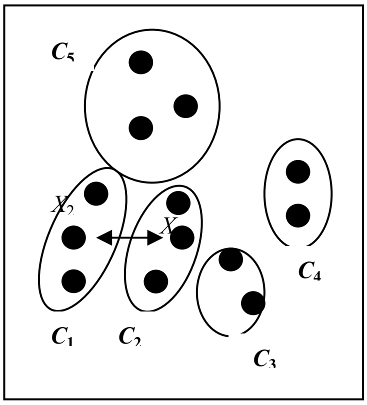

Figure 1 (b): Semi-supervised clustering *after* user feedback

User feedback-driven clustering technique as described in [3] uses three phases to cluster a set of documents:
Phase 1: Pre-clustering (generation of fine-grained clusters): the system partitions a given document collection into small clusters based on the distance between documents. The complete-linkage hierarchical agglomerative clustering is used as a basic clustering algorithm to generate fine-grained clusters.

Phase 2: Supervision phase (user feedback): in this phase, two types of document bundles (i.e. group of documents) are created: a positive bundle and a negative bundle. These bundles are developed by a relevance feedback program from interviews with the user. The interview program starts by extracting a set of documents randomly from a set of pre-clusters, and then the user determines relevant documents to put them in the positive bundle and irrelevant documents to put them in the negative bundle.

Phase 3: Re-clustering (assigns each of the pre-clustered document to its nearest positive document bundle): in this phase each of the pre-clustered documents is assigned to the positive bundle in which its nearest document is found. During assigning of pre-clusters to the positive bundle; the local centroid of the cluster is updated. At the same time the negative bundle is observed to prevent assigning any documents in the negative bundle to the positive bundle. If such document is assigned to the positive bundle then it is re-assigned to another cluster that has its second nearest document.

3. A User-Enhanced Clustering Technique

This section demonstrates in details the proposed work. It uses three phases, namely: Creating Initial Clusters, Filtering, and Re-Clustering.

Creating Initial Clusters:

To demonstrate the effectiveness of user browsing time as a means of improving cluster quality, a clustering algorithm to create initial clusters is needed. The researchers choose Frequent Item-set based Hierarchical Clustering (FIHC) algorithm to create the initial clusters [1]. The FIHC was selected based on the following reasons:

1. This document clustering algorithm produced consistently high quality clusters.
2. As shown in [1] it gives the best results when it is compared with other document clustering algorithms.
3. It could be applied to a large and complicated data set (like web pages).
4. Its output is an XML file, which can be easily converted into other data formats.

FIHC was modified to satisfy this research requirement; in particular the modification was in its output to obtain document vectors and global frequent items which are needed in the re-clustering phase of the proposed technique.

Filtering:

The main idea of the proposed technique is to derive user feedback from user browsing time. The assumption is that when a user spends a long time in viewing a web page, it means that this web page is relevant to the user's query. In contrast, if a user spends a short time in viewing a web page, it means that this web page is irrelevant to that user.

To get user browsing time for each web page which is returned as a result to user query, software called "Genius Filter" was prepared, it is used as a search engine and as observer to the user browsing time. Genius Filter has an interface where the user can type his query and as a result, a set of web pages is returned.

The user writes his query, which must be a cluster label, and starts the searching process, the software searches in the clusters about a cluster label that matches the user's query and returns a list of links to web pages found in this cluster. When the user clicks on any link, the target web page opens in a browser window. The timer starts when the web page is completely loaded. To gain a high accurate user browsing time, the timer stops when a web page is minimized or is inactive because it isn't currently being browsed by the user.

Several users were trained on using the Genius Filter and were allowed to submit queries to this filter. For every query and for every returned page (that was opened by the user) the browsing time was recorded in a database. If several users open the same page, then the browsing time for that page is the sum of all browsing times.

After a period of browsing web pages (at least one week) the average browsing time is calculated and any web page that has average browsing time less than a specified threshold (determined experimentally) is considered irrelevant, otherwise, it is considered relevant to the user's query.

Figure 2 part (a) shows the first phase of the proposed methodology; C_1, C_2, and C_3 are the initial clusters and solid circles represent documents, Figure 2 part (b) shows the status of the initial clusters after the filtering phase; solid squares represent irrelevant web pages based on their average user browsing time which didn’t achieve the threshold. All irrelevant web pages were moved to a special cluster called *irrelevant-cluster*.

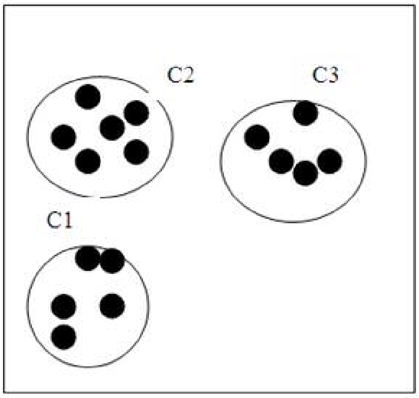

Figure 2 (a): First Phase of the proposed technique.

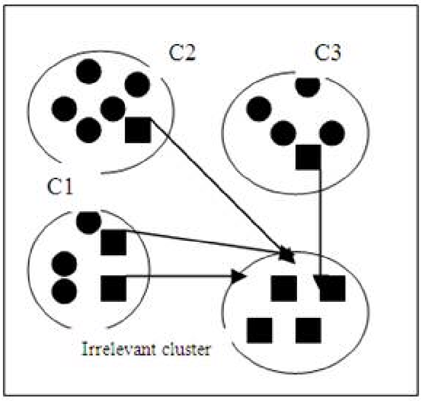

Figure 2 (b): Status of the clusters after the filtering phase

Re-clustering:

The purpose of the third phase of this technique is to determine the best cluster for each irrelevant web page. The rank of relevant web pages inside clusters based on user browsing time was investigated. The top *N* relevant web pages in each cluster are used to represent their clusters in the re-clustering phase.

The top *N* relevant web pages from each cluster were taken and combined in one group as one data set, and then the *K* Nearest Neighbor (*K*-NN) algorithm [5] was applied to determine the nearest *K* web pages in the data set to each irrelevant web page. The cluster which contains most of the *K* nearest web pages is considered as the best cluster of the irrelevant web page.

4. Experimentation and Result analysis

4.1 Dataset and Experiments

To evaluate the proposed technique, the researchers have pre-specified that the dataset will consist of 5 classes which are *TOEFL*, *Islam*, *Java*, *Computer*, and *Sport*. To create the data set, a collection of URLs relevant to these five topics is created by using Visual Web Spider [8]. After filling these URLs into an MS Access database, *Genius Downloader* reads each URL from the database and saved its target web page on the hard disk; the last step is concerned with converting each web page into a document (i.e. removing images and multimedia controls) this was done by using Total HTML Converter [9]. The number of collected pages per class is shown in Table 1.

Class Label	Number of Documents
TOEFL	191
Islam	125
Java	100
Computer	109
Sport	233
Total	**758**

Table 1: Distribution of pages to classes

Experiments have been done using 30 Pentium IV client PCs and one PC was used as a database server. The operating system used was Windows XP. Also .NET Framework was installed on client PCs to enable them to run the (Genius Filter). On the server side, MS SQL Server 2000 Enterprise Edition was installed.

The experiment starts by running Frequent Itemset-based Hierarchal Clustering (FIHC) program on the data set to get the initial clusters. Table 2 shows the distribution of web pages to clusters. Cluster labels were determined manually to match original class labels, these labels were determined by counting the documents that belong to the same original class, and then the cluster label is set to the class name that most of the cluster's documents belong to it. For example, given a cluster "C"; its label will be TOEFL if it contains many documents that talk about TOEFL. After creating the initial clusters by using the FIHC algorithm, the result was converted from an XML file to a Relational Database; also document vectors, frequent global items and cluster frequent items were stored in the database.

Cluster Label	Number of Documents
TOEFL	**597**
Islam	**68**
Java	**37**
Computer	**29**
Sport	**27**

Table 2: Distribution of documents to clusters after applying the FIHC algorithm.

After creating the initial clusters and populating the database, the filtering phase starts. In this phase, (Genius Filter) was run for a duration of two weeks on 30 PCs. During this period, the number of visits achieved was 3642 for all web pages in all clusters. The developed software stored user browsing time for each web page. After the two weeks have passed, the average browsing time was calculated for each web page as follows:

$$\text{Average browsing time} = \frac{\text{Total browsing time}}{\text{number of visitors}}$$

The software requires a pre-specified threshold. This value is used to determine relevant from irrelevant pages. The main issue in choosing this threshold is that it must raise precision in clusters to emphasize that each irrelevant web page is discovered. Five different values as thresholds have been used: 10, 20, 30, 40, and 50 seconds but the experiments show that 40 seconds is the best threshold because it gives the highest precision.

At the end of this phase, a high precision is expected in all clusters because all irrelevant web pages were discovered and removed. In contrast, low recall is expected in all clusters because irrelevant web pages are not re-assigned to their proper clusters. So, to determine if this phase is a success the focus was on precision before and after this phase.

The re-clustering phase comes after the filtering phase. In this phase both precision and recall were analyzed; so we asses clusters quality based on the F-measure. The re-clustering phase is done firstly as described in Section 3, and then an optimization that investigates the URL address of the web page to determine its best cluster was utilized. This approach is simple and it says that if an URL address of the web page contains any cluster label then this web page is re-clustered to that cluster, if there are more than one cluster label in the URL address of the web page, then the web page is re-clustered by using the KNN algorithm as described in section 3.

4.2 Result Analysis

Figure 3 shows the quality of the initial clusters. Low quality is demonstrated in all initial clusters. This result reflects the weakness of document clustering techniques which are used to cluster web pages with regard to the fact that the researchers have used FIHC which is one of the best document clustering algorithms as described in [12].

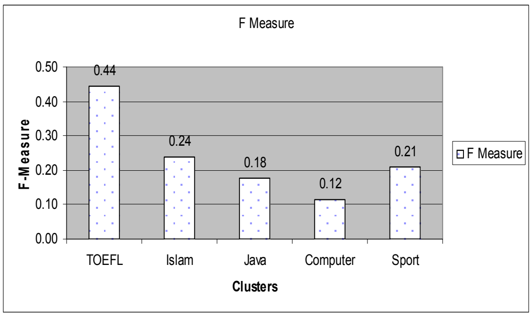

Figure 3: F-measure values of the initial clusters

Figure 4 shows the clusters precision after the filtering phase. It shows a high precision of all clusters because most of the irrelevant web pages were discovered and removed.

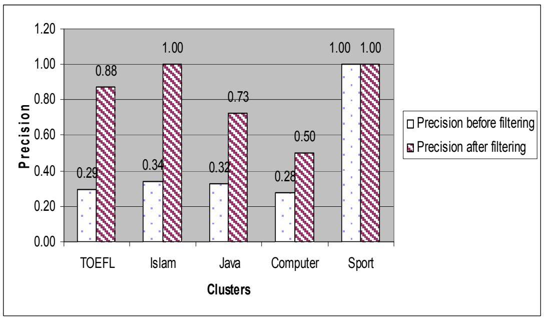

Figure 4: Precision of the clusters before and after filtering

Figures 5, 6, and 7 show the enhancement of clusters quality after the re-clustering phase with different values of *K* and *N* (used in KNN algorithm) without optimization. The best enhancement occurred when *K*=5 and *N*=8 as shown in Figure 8 which compares the average of F-measure values in all clusters with different *K* and *N* values.

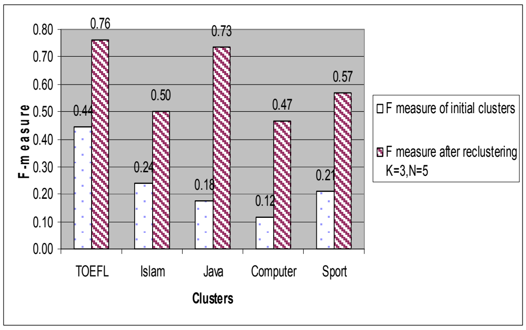

Figure 5: Enhancements of clusters' quality after re-clustering with *K*=3 and *N*=5

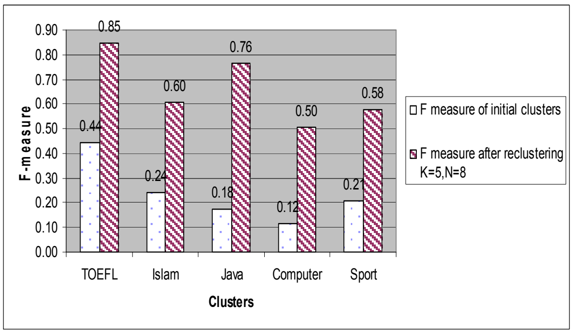

Figure 6: Enhancements of clusters' quality after re-clustering with *K*=5 and *N*=8

To view the efficiency of the proposed optimization technique, which employs the URL of a webpage to deduce its cluster label, the best quality of the clusters after re-clustering phase without optimization (at K=3 and N=5) was compared with re-clustering with optimization as shown in Figure 9. The comparison shows that the optimization technique enhances the quality in all clusters.

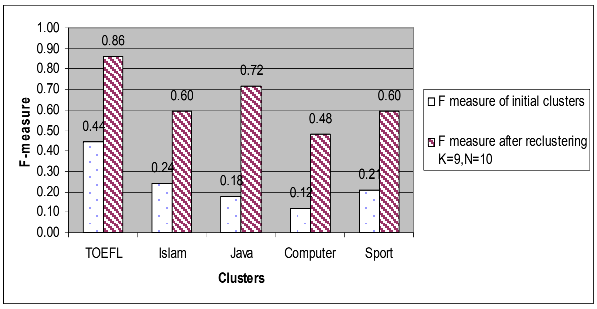

Figure 7: Enhancements of clusters' quality after re-clustering with *K*=9 and *N*=10

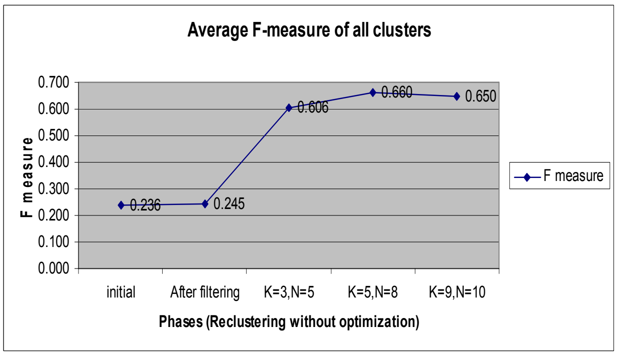

Figure 8: A comparison between average F-measure values with different *K* and *N* values.

Figure 10 shows the comparison between the quality of the initial clusters and the quality of the clusters after re-clustering with optimization at the best K and N values which reflect the enhancement degree after using the technique proposed in this paper.

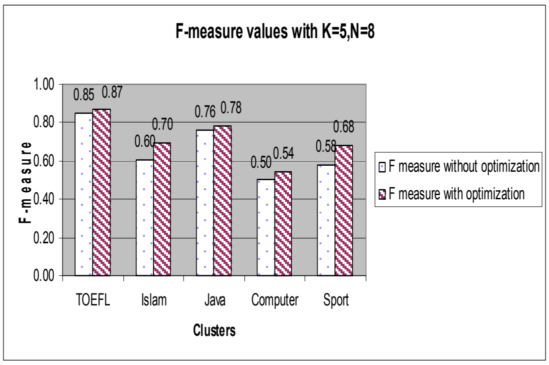

Figure 9: A comparison of F-measure values at *K*=5 and *N*=8 with and without optimization

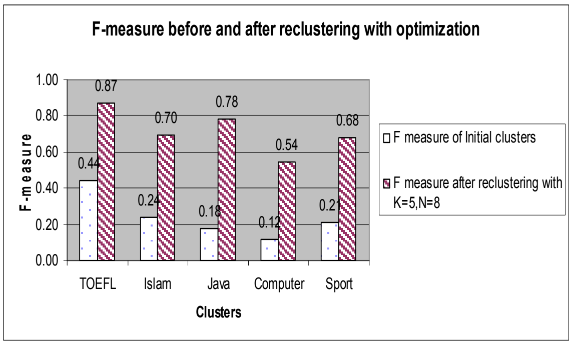

Figure 10: Results of clusters' quality enhancement

5. Conclusions and Future Work

This research contributes a new technique to enhance clusters' quality and web search precision; it depends on deriving user feedback from user browsing time and also using user browsing time to determine the most relevant web pages which will be used to re-cluster irrelevant web pages. The current technique has several new features namely:

1. User feedback is gathered implicitly (i.e. without the knowledge of the user).
2. It is generic; as it can be used as part of any clustering algorithm to enhance the quality of returned clusters.
3. It can be added to offline clustering algorithms or online clustering algorithms.

In the current version of the developed software, pages that remain in the irrelevant-pages cluster after the re-clustering phase are not further processed (i.e. the software does not attempt to infer their cluster label). This is because the current version deals with a closed set of cluster labels; therefore, any page that does not belong to any of the predefined cluster labels is left in the irrelevant-pages cluster. The researchers plan to extend the software to treat such cases, and consider other types of implicit user feedback.

References

1. B. C. M. Fung, K. Wang, and M. Ester. "Hierarchical Document Clustering Using Frequent Itemsets". In Proc. of the 3rd SIAM International Conference on Data Mining (SDM 2003), pages 59-70, San Francisco, CA, USA, May 1-3, 2003.
2. David Cohn, Rich Caruana. "Semi-Supervised Clustering: Incorporating User Feedback to Improve Cluster Utility", American Association for Artificial Intelligence 2000. Available from URL http://www.aaai.org.
3. Han-joon Kim ; Sang-goo Lee, "User-Feedback Driven Document Clustering Technique for Information Organization", Ieice Transactions on Information and Systems 2002; vol. E85-D(6): 1043-1048.
4. Jeffrey W. Seifert, "CRS Report for Congress. Data Mining: An Overview", Congressional Research Service, Congress library, Updated June 7, 2005.
5. Jiawei Han, Micheline Kamber, "Data Mining: Concepts and Techniques", Morgan Kaufmann, 2001.
6. Ryen W. White, Ian Ruthven, Joemon M. Jose, C. J. Van Rijsbergen, "Evaluating Implicit Feedback Models Using Searcher Simulations", ACM Transaction on Information Systems 2005; 23(3): 325-361.
7. Steve Fox, Kuldeep Karnawat, Mark Mydland, Susan Dumais, Thomas White. "Evaluating Implicit Measures to Improve Web Search", ACM Transactions on Information Systems 2005; 23(2): 147-168.
8. http://www.newprosoft.com
9. http://www.coolutils.com