

## A Survey On Web Search Result Clustering And Engines

Poonam B.Lohiya

Lecturer, Information Technology

PRMITR,,Amravati

### **Abstract:**

Now a days World Wide Web is a very large distributed digital information space. The ability to search and retrieve information from the Web efficiently and effectively is an enabling technology for realizing its full potential. Current search tools retrieve too many documents, of which only a small fraction are relevant to the user query. Web clustering engines organize search results by topic, thus offering a complementary view to the flat-ranked list returned by conventional search engines. This paper highlights the main characteristics of a number of existing Web clustering engines and also discuss how to evaluate their retrieval performance.

**Keywords:** Information Retrieval; Search Engine; Document Clustering.

### **1. Introduction:**

The information available on the web is unstructured, disorganized, dynamic and heterogeneous in nature and enormously large. Moreover the process of retrieval is highly affected by the vague query put up by the average user. Although today's search engines are smarter than earlier, ambiguous queries are still a major problem. To answer all the possible meaning of an ambiguous query, search engines return too many results which are not necessarily relevant to the user's need. Usually user has to traverse several search result pages to get to the desired result. A way of assisting users in finding what they are looking for quickly is

to group the search results by topic. The user does not have to reformulate the query, but can merely click on the topic most accurately describing his or her specific information need. This grouping of result is called Clustering. More specifically, it is a process of grouping similar documents into clusters so that documents of one cluster are different from the documents of other clusters. There are many web clustering engines available on the web (Carrot2, Vivisimo, SnakeT, Grouper etc) which give the search results in forms of clusters. A web clustering engine takes the result, returned by the search engine as input and performs clustering and labeling on that result. This process is usually seen as complementary rather than alternative and different to the search engine. The main use for web search result clustering is not to improve the actual ranking, but to give the user a quick overview of the results. One common feature of most current clustering engines is that they do not maintain their own index of documents; similar to meta search engines, they take the search results from one or more publicly accessible search engines. The low precision of the web search engines coupled with the long ranked list presentation make it hard for users to find the information they are looking for. It takes lot of time to find the relevant information. Typical queries retrieve hundreds of documents, most of which have no relation with what the user was looking for. According to this, we considered Web-snippet clustering engine is a useful complement to the flat, ranked list of results offered by classical search engines (like Google). Web snippet (short description) clustering also known as Web Search Results Clustering is an attempt to apply the idea of clustering to snippets returned by a

search engine in response to a query.[5] Thus, it can be perceived as a way of organizing the snippets into set of meaningful thematic groups. Actually, clustering engines are usually seen as complementary instead of alternative to search engines. Not only has search results clustering attracted considerable commercial interest, but it is also an active research area, with a large number of published papers discussing specific issues and systems. Search results clustering is clearly related to the field of document clustering but it poses unique challenges concerning both the effectiveness and the efficiency of the underlying algorithms that cannot be addressed by conventional techniques. The main difference is the emphasis on the quality of cluster labels, whereas this issue was of somewhat lesser importance in earlier research on document clustering. A clustering engine tries to address the limitations of current search engines by providing clustered results as an added feature to their standard user interface and meaningful labels.



Figure 1:Carrot2 Clustering Engine

In Figure 1, a snapshot of the Carrot2 clustering engine clearly shows how an ambiguous word like “concord” relates to different groups. From the left pane of the

engine, a user can easily locate what he/she is actually searching, whether it is a record label or a town in New Hampshire. Carrot2 and other clustering engines does far better than the average search engine, still there is a need for an efficient and effective clustering engine which is cost effective in terms of time as compared to traditional search engines.[6]

## 2. Web Clustering engines goal:

Plain search engines are usually quite effective for certain types of search tasks, such as navigational queries (where the user has a particular URL to find) and transactional queries (where the user is interested in some Web-mediated activity). However, they can fail in addressing informational queries (in which the user has an information need to satisfy), which account for the majority of Web searches [7]. This is especially true for informational searches expressed by vague, broad or ambiguous queries. A clustering engine tries to address the limitations of current search engines by providing clustered results as an added feature to their standard user interface. We emphasize that clustering engines are usually seen as complementary—rather than alternative—to search engines. In fact, in most clustering engines the categories created by the system are kept separated from the plain result list, and users are allowed to use the list in the first place. The view that clustering engines are primarily helpful when search engines fail is also supported by some recent experimental studies of Web searches [2].

The search aspects where clustering engines can be most useful in complementing the output of plain search engines are the following.

- *Fast subtopic retrieval.* If the documents that pertain to the same subtopic have been correctly placed within the same cluster and the user is able to choose the

right path from the cluster label, such documents can be accessed in logarithmic rather than linear time.

- *Topic exploration.* A cluster hierarchy provides a high-level view of the whole query topic including terms for query reformulation, which is particularly useful for informational searches in unknown or dynamic domains.
- *Alleviating information overlook.* Web searchers typically view only the first result page, thus overlooking most information. As a clustering engine summarizes the content of many search results in one single view on the first result page, the user may review hundreds of potentially relevant results without the need to download and scroll to subsequent pages.

### 3. Web Search Result Clustering:

Clustering of web search results has been studied in the area of Information Retrieval (IR). The goal of clustering search result is to give user an idea of what the result contains. This idea is in the form of clusters. Clustering in context of web search result means organizing query result pages into groups based on their similarity between each other. Vivisimo, Carrot2, Kartoo etc are some of common commercial clustering engines available. Search result clustering techniques specific to the search engine result can be broadly classified as content-based and topology-based clustering. [5]

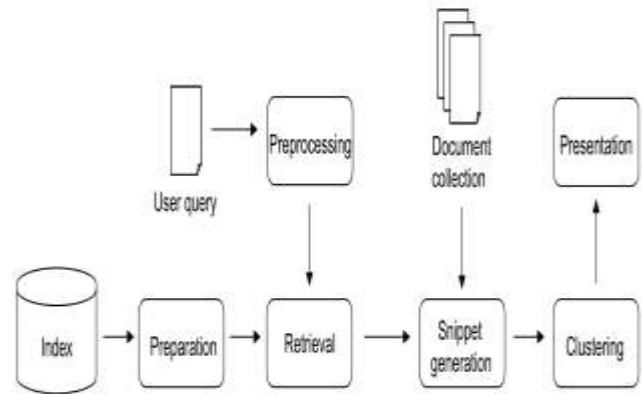


Figure 2: A generic Web search result clustering system

### 4. Architecture of Web Search Engine:

The Search Engine component is a part of the Information Retrieval model component. Its main responsibility is the comparison of documents based on their document models through obtaining documents similarity values. In today's search engines, Clustering of results is the next step up from ranking of documents Web search engines did not come into existence until 1994.

A search engine has four components:

- Document processor indexes new documents. Indices are a mapping between words and what documents they appear in. Most engines are spider-based, so a crawl of the web for new documents and the updating of the index is automated.
- Query processor inspects a user's query and translates it into something internally meaningful.
- Matching function uses the above internally meaningful representation to extract documents from the index.
- Ranking scheme positions the more-relevant documents on top, using some relevance measure.

We may classify web search engines according to the set of features they explore [7]. First generation web search engines, starting in 1994 with WebCrawler and Lycos, explore on-page data (content and

formatting). They support mostly informational queries. The second generation, emerging in 1998, with Google (Page Rank), uses off-page web specific data (link analysis, anchor text and click streams data) and supports both informational and navigational queries. The third generation, appearing during the first years of 2000 attempts to merge multiple sources of evidence and aims to support all kinds of queries. Modern Web IR is a discipline which has exploited some of the classical results of Information Retrieval developing innovative models of information access. Therefore, search engines have established as a revolutionary working metaphor. If someone needs information about a book, an address, a research paper, a flight ticket, or almost any other topic, they just make a query on a search engine. In this paragraph we briefly review the architecture of a typical search engine. The architecture of a search engine is given in Figure 3. Crawlers are distributed agents which gather information from the Web. They crawl the Web graph visiting the pages according to some policies (BFS, DFS, random, topic focused, prioritized) and store the pages in a local Page Repository. From time to time, the pages are indexed and analyzed to produce a fast index able representation of both the documents and the link structures. Both the textual and structural indexes are then analyzed to rank the documents stored in the repository. For efficiency reasons, part of this ranking process can be performed off line, before the query is submitted through the query engine. Nowadays, modern search engines index billions of objects on distributed platforms of thousands of commodity PCs running Linux. The index is often organized in a two tier structure and is replicated by thousands of autonomous clusters of servers. This distribution is needed to sustain the peak load of thousands of queries per second.

Users communicate with the query processor, which is the only visible component. It carries out several tasks, usually (but not limited to):

- Tokenizing of the query to remove invalid characters, and to recognize meta-keywords or special syntactic operators.
- Removal of stop words; words which are too common and rarely help in the search (e.g. the, a, of, to, which).
- Stemming; a process designed to improve the performance of IR systems, involving normalizing semantically similar words to their root forms (e.g. produce, produced, producer, producers, produces and producing map to produce-).
- Assigning a weight to each keyword/key phrase, to aid with ranking [3].

After results are retrieved by the matching function, they are ranked by relevance based on some ranking measure and set of heuristics (called the ranking scheme). Often taken into account are:

- Term frequency how many times keywords appear in the document.
- Inverted document frequency a value which aims to determine how important a term is in discriminating a document from others.[3]
- Semantic proximity words synonymous to a given keyword may be matched, boosting the score of the document.
- Term position keywords appearing in the title or heading (rather than the body) should contribute more to a document's weight.
- Term proximity a document in which the query terms are close together is considered more relevant than one in which they are far apart.
- Cluster distance how far apart groupings of matched terms are.
- Percentage of query terms matched.[3][5]



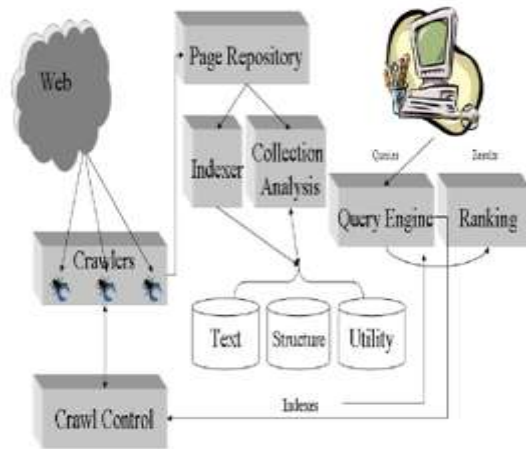


Figure 3: Architecture of Web Search Engine

### 5. Performance Evaluation:

Lancaster and Fayen (1973) once listed 6 criteria for assessing the performance of information retrieval systems.

They are: 1) Coverage, 2) Recall, 3) Precision, 4) Response time, 5) User effort, and 6) Form of output.

Although the criteria were set up more than two decades ago and a great deal has been done to reduce user effort (e.g., design friendly user interface) in using the system, they still seem quite applicable to evaluating information retrieval systems today.

Based on our knowledge and experience gained from the current study, we believe that one needs to consider the following aspects when evaluating a Web search engine.[11]

- **Composition of Web Indexes**

Whenever a Web search request is issued, it is the web index generated by Web robots or spiders, not the web pages themselves that has been used for retrieving information. Therefore, the composition of Web indexes affects the performance of a Web search engine. There are three components that the authors would like to inspect regarding the makeup of a Web index, namely, coverage,

update frequency and the portions of Web pages indexed (e.g., titles plus the first several lines, or the entire Web page). We understand that the magnitude of all three components depends largely on the power and sophistication of the hardware and software that make the Web index or database. On the other hand, larger coverage, frequent updates and full text indexing do not necessarily mean better Web search engines in other measurements.

- **Search Capability**

A competent Web search engine must include the fundamental search facilities that Internet users are familiar with, which include Boolean logic, phrase searching, truncation, and limiting facilities (e.g., limit by field).

- **Retrieval Performance**

Retrieval performance is traditionally evaluated on three parameters: precision, recall and response time. While the three variables can all be quantitatively measured, extra caution should be exercised when one judges the relevance of retrieved items and estimates the total number of documents relevant to a specific topic in the Web system.

- **Output Option**

This evaluation component should be examined from two perspectives. One is the number of output options a Web search engine offers, whereas the other deals with the actual content of the output. Sometimes, one search engine may appear quite impressive in one aspect, but in reality it cannot satisfy its users because of its weakness in the other facet of this evaluation criterion.

- **User Effort**

User effort refers to documentation and interface in this study. Well-prepared documentation and a user-friendly interface play a notable role in users' selection of Web search engines. Since there are more than two dozen of them available, the

attractiveness of each Web search engine is expressed, to its users, mainly in its documentation and interface.[5]

## 6. Conclusion:

In this paper we conclude that general users put short, ambiguous queries which can't specify the actual information need of the users. Clustering is the best possible solution for this problem, it facilitates quick browsing throughout the search result. To improve the search result clustering, First, more work needs to be done to improve the quality of the cluster labels and the coherence of the cluster structure. Second, the incrementality, because the web pages change very frequently and because new pages are always added to the web. Third, the fact that very often a web page relates to more than one subject should also be considered and lead to algorithms that allow for overlapping clusters. Fourth, Inconsistency is another problem. The contents of a cluster do not always correspond to the label and the navigation through the cluster sub hierarchies does not necessarily lead to more specific results. Fifth, advanced visualization techniques might be used to provide better overviews and guide the interaction with clustered results.

## 7. References:

- [1] Carpenito C, Osinski S, Romano G, and Weiss D (2009) A Survey of Web Clustering Engines. ACM Computing Surveys, Vol. 41, No. 3, Article 17.
- [2] A Survey of Web Clustering Engines, Claudio Carpineto Fondazione Ugo Bordoni Stanislawosin´ski Carrot Search Giovanni Romano Fondazione Ugo Bordoni and Dawid Weiss ,Poznan University of Technology.
- [3] Salton, G. (1989), Automatic Text Processing: the Transformation, Analysis,

and Retrieval of Information by Computer, Addison-Wesley Longman Publishing Co., Inc.

- [4] Han J and Kamber M (2001) Data Mining -Concepts and Techniques. Academic Press.
- [5] R. Subhashini et. al. / Indian Journal of Computer Science and Engineering Vol. 1 No. 4 392-401
- [6] Kishwar Sadaf1 and Mansaf Alam2, International Journal of Computer Science & Engineering Survey (IJCSSES) Vol.3, No.4, August 2012
- [7] Broder, A. (2002) A taxonomy of web search. SIGIR Forum. 36:2. p. 3-10
- [8] Cutting, D.R., Karger, D.R., Pedersen, J.O., Tukey, J.W. 1992. Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections. Proc. ACM SIGIR Conference on Research and Development in Information Retrieval, pp.318-329
- [9] E. Selberg and O. Etzioni, Multi-service search and comparison using the MetaCrawler, in: Proceedings of the 4th World Wide Web Conference (WWW4), 1995.
- [10] Ferriagina, P. and Gulli, A. 2004. The Anatomy of SnakeT: A Hierarchical Clustering Engine for Web Page Snippets.
- [11] Lancaster, F.W., and Fayen, E.G. (1973). Information Retrieval On-Line Los Angeles, CA: Melville Publishing Co. Chapter 6.

## Author Profile:

I am Ms.Poonam B.Lohiya received my B.E Degree in Information Technology from H.V.P.M College of Engineering and Technology in the year 2011 and M.E (Applied Computer Science and Engineering) from PRMCEAM,Amravati in 2012.

I am Working as a Lecturer in Information Technology dept,PRMITR,Amravati.