Automatic Link Generation for Search Engine Optimization

Reyner D’souza, Apurva Kulkarni, and Imran Ali Mirza

***Abstract*—Traditional text search engines accomplish document retrieval by taking a query from the user, and then returning a set of documents matching the user’s query. A web search engine often returns thousands of pages in response to a broad query. This makes it very difficult for users to browse or to identify relevant information from the results returned. In order to retrieve the documents of interest, the user must formulate the query using the keywords that appear in the documents. This is a difficult task, if not impossible, for ordinary people who are not familiar with the vocabulary of the data corpus. Clustering methods can be used to automatically group the retrieved documents into a sorted list of meaningful categories by analyzing the results for related content.**

***Index Terms*—Search engine, PageRank, Automatic Link Generation, Web-site clustering.**

I. Introduction

When a query is entered in a search engine many pages are returned in form of result. Pages are returned on the basis of their page rank. Higher the rank of the page, higher is the chances of it to be found on the top in relation with the search term or the query. The rank of the page depends on algorithms like the PageRank algorithm and many other factors of the web page like no of out links, bounce back rate and other such factors. It may be possible that the query exists in a page which has high PageRank for its content but that data may not be relevant to the user.

It has also been observed that a search term may result into multiple results that may be related to various fields. The best example can be that if a name of the person is searched then the term may refer to the name of an organization, an executable application, an well known person or just a local search for some student or person. In such a case the data provided to a search engine is inadequate and the data returned to the person may be wrong or irrelevant. The user will have to search latter pages which have a lower rank. Google has devised a better method for entering the search term is that of real time suggestions of Terms based on that of most frequently searched items in relation to search terms.

II. Main Content

A. Current Search Engines

It has been observed that most of the times, especially in case of local searches that the search engine does not return the data relevant to the search term. Whenever a web user enters a query in a search engine many pages are returned in form of result. These pages are returned on the basis of their page rank. Higher the rank of the page, higher is the chances of it to be found on the top in relation with the search term or the query. The rank of the page depends on algorithms like the PageRank algorithm and many other factors of the web page like no of out links, bounce back rate and other such factors. It may be possible that the query exists in a page which has high PageRank for its content but that data may not be relevant to the user. Other places it has been observed that a search term may result into multiple results that may be related to various fields. The best example can be that if a name of the person is searched then the term may refer to the name of an organization, an executable application, a well known person or just a local search for some student or person. In such a case the data provided to a search engine is inadequate and the data returned to the person may be wrong or irrelevant. In such a case the user will have to search latter pages which have a lower rank. Google has devised a better method for entering the search term is that of real time suggestions of Terms based on that of most frequently searched items in relation to search terms.

B. PageRank (PR)

PageRank is a numeric value that represents how important a page is on the web. Google Page Rank is Google's assessment of the importance and popularity of a website based on the number and quality of inbound links from other websites on the World Wide Web. Google figures that when one page links to another page, it is effectively casting a vote for the other page. The more votes that are cast for a page, the more important the page must be. Also, the importance of the page that is casting the vote determines how important the vote itself is. Google calculates a page's importance from the votes cast for it. How important each vote is is taken into account when a page's PageRank is calculated. PageRank is Google's way of deciding a page's importance. It matters because it is one of the factors that determine a page's ranking in the search results. It isn't the only factor that Google uses to rank pages, but it is an important one.

C. Web Document Clustering

Document clustering involves the use of descriptors and descriptor extraction. Descriptors are sets of words that

Manuscript received May 22, 2012; revised June 15, 2012.

Reyner D’souza and Apurva Kulkarni are with Don Bosco Institute of Technology, Mumbai, India (e-mail: rynrdsouza@gmail.com).

Imran Ali Mirza is with the Department of Computer Engineering, Don Bosco Institute of Technology, Mumbai, India.

describe the contents within the cluster. Document clustering is generally considered to be a centralized process. Examples of document clustering include web document clustering for search users. Web document clustering has been traditionally investigated mainly as a means of improving the performance of search engines by pre-clustering the entire corpus. However, clustering has also been investigated as a post-retrieval document browsing technique. Numerous documents clustering algorithms can be used to cluster the documents.

1) *K-means Clustering*

Linear time clustering algorithms are the best candidates to comply with the speed requirement of online clustering.These include the K-Means algorithm - O(nkT) time complexity where k is the number of desired clusters and T is the number of iterations , and the Single- Pass method - O(nK) were K is the number of clusters created. One advantage of the K-Means algorithm is that, it can produce overlapping clusters. Its chief disadvantage is that it is known to be most effective when the desired clusters are approximately spherical with respect to the similarity measure used. There is no reason to believe that documents (under the standard representation as weighted word vectors and some form of normalized dot-product similarity measure) should fall into approximately spherical clusters. The Single-Pass method also suffers from this disadvantage, as well as from being order dependant and from having a tendency to produce large clusters. It is, however, the most popular incremental clustering algorithm.

2) *Suffix Tree clustering*

Suffix Tree Clustering (STC) is a linear time clustering algorithm that is based on identifying the phrases that are common to groups of documents. A phrase in our context is an ordered sequence of one or more words. We define a base cluster to be a set of documents that share a common phrase.

STC has three logical steps:

1) Document “cleaning”,
2) Identifying base clusters using a suffix tree, and
3) Combining these base clusters into clusters.

3) *Lingo clustering*

The majority of open text clustering algorithms follows a scheme where cluster content discovery is performed first, and then, based on the content, the labels are determined. But very often intricate measures of similarity among documents do not correspond well with plain human understanding of what a cluster’s “glue” element has been. To avoid such problems Lingo reverses this process, we first attempt to ensure that we can create a human-perceivable cluster label and only then assign documents to it. Specifically, we extract frequent phrases from the input documents, hoping they are the most informative source of human-readable topic descriptions. Next, by performing reduction of the original term document matrix using SVD, we try to discover any existing latent structure of diverse topics in the search result. Finally, we match group descriptions with the extracted topics and assign relevant documents to them.

III. Working Model

Many document clustering algorithms rely on off-line clustering of the entire document collection, but the Web search engines’ collections are too large and fluid to allow off-line clustering. Therefore clustering has to be applied to the much smaller set of documents returned in response to a query. Because the search engines service millions of queries per day, free of charge, the CPU cycles and memory dedicated to each individual query are severely curtailed. Thus, clustering has to be performed on a separate machine, which receives search engine results as input, creates clusters and presents them to the user. Hence in order to obtain query results, external search-engines are used as data-source. In the working model different document sources like bing, google are used to obtain results for a query. This eliminates the installation of a separate Web search engine.

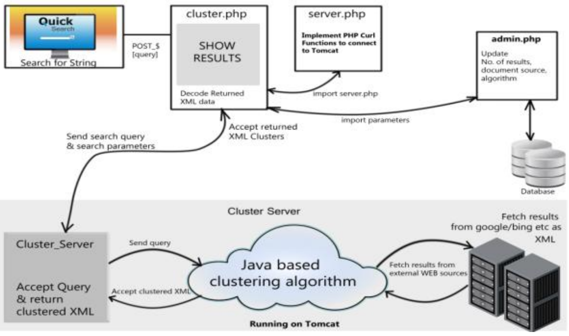


Fig. 1. Entering of search term.

The working model can automatically cluster small collections of documents, e.g. search results or document abstracts, into thematic categories

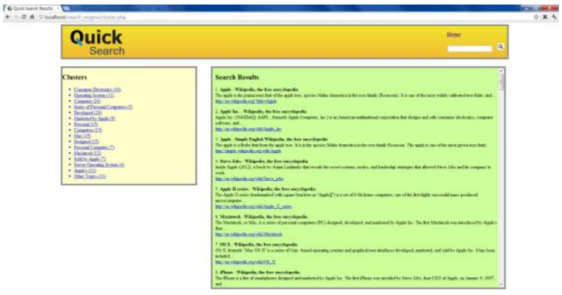
It works by accepting a search query from the user. As seen in Fig. 1. The query along with the search parameters i.e. number of search results, algorithm, and document source is is passed to the clustering server. The parameters are fetched from a mySQL database. The clustering server runs a Java based clustering algorithm running on Apache Tomcat. The clustering server fetches results for the search query from a external document source like bing or google search. Other document sources can also be integrated. The search results are accepted in form of an XML document. In order to improve the performance, only a limited number of search results are fetched.

Quick Search
localhost/search_engine/index.html
Quick
Search
Designed by Reyner D'souza & Apurva Kulkarni

Fig. 2. Entering of search term

The selected clustering approach is applied on the XML document. The algorithm divides the results into clusters, using descriptors & descriptor extraction. Descriptors are sets of words that describe the contents within the cluster. After forming the clusters, the documents are divided &

assigned to the nearest cluster. The search results and the clusters are returned as an XML output data. The XML data is accepted & parsed by a PHP script implementing cURL. cURL is used to connect and communicate to different types of servers with different protocols. The XML is processed to obtain the search results & the document clusters. The search results and the clusters are displayed to the user as shown in Fig. 3.



Clusters

- Consumer Electronics (10)
- Operating System (11)
- Computer (24)
- Series of Personal Computers (5)
- Developed (19)
- Marketed by Apple (9)
- Personal (15)
- Computers (15)
- Mac (15)
- Designed (15)
- Personal Computer (7)
- Macintosh (12)
- Sold by Apple (7)
- Server Operating System (4)
- Apple's (11)
- Other Topics (33)

Search Results

1. Apple - Wikipedia, the free encyclopedia
The apple is the pomaceous fruit of the apple tree, species Malus domestica in the rose family (Rosaceae). It is one of the most widely cultivated tree fruits, and ...
http://en.wikipedia.org/?title=Apple

2. Apple Inc. - Wikipedia, the free encyclopedia
Apple Inc. (NASDAQ: AAPL; formerly Apple Computer, Inc.) is an American multinational corporation that designs and sells consumer electronics, computer software, and ...
http://en.wikipedia.org/wiki/Apple_inc

3. Apple - Simple English Wikipedia, the free encyclopedia
The apple is a fruit that from the apple tree. It is in the species Malus domestica in the rose family Rosaceae. The apple is one of the most grown tree fruits.
http://simple.wikipedia.org/wiki/Apple

4. Steve Jobs - Wikipedia, the free encyclopedia
Inside Apple (2012), a book by Adam Lashinsky that reveals the secret systems, tactics, and leadership strategies that allowed Steve Jobs and his company to work.
http://en.wikipedia.org/wiki/Steve_Jobs

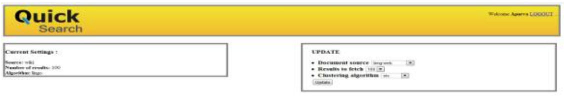
5. Apple II series - Wikipedia, the free encyclopedia
The Apple II series (trademarked with square brackets as "Apple][") is a set of 8-bit home computers, one of the first highly successful mass-produced microcomputer ...
http://en.wikipedia.org/wiki/Apple_II_series

6. Macintosh - Wikipedia, the free encyclopedia
The Macintosh, or Mac, is a series of personal computers (PC) designed, developed, and marketed by Apple Inc. The first Macintosh was introduced by Apple's then ...
http://en.wikipedia.org/wiki/Macintosh

7. OS X - Wikipedia, the free encyclopedia
OS X, formerly "Mac OS X" is a series of Unix - based operating systems and graphical user interfaces developed, marketed, and sold by Apple Inc. It has been included ...
http://en.wikipedia.org/wiki/OS_X

8. iPhone - Wikipedia, the free encyclopedia
The iPhone is a line of smartphones designed and marketed by Apple Inc. The first iPhone was unveiled by Steve Jobs, then CEO of Apple, on January 9, 2007, and ...

Fig. 3. Classification and clustering of results



Current Settings :

Source: wiki
Number of results: 100
Algorithm: lingo

UPDATE

- Document source [illegible]
- Results to fetch 100
- Clustering algorithm stc

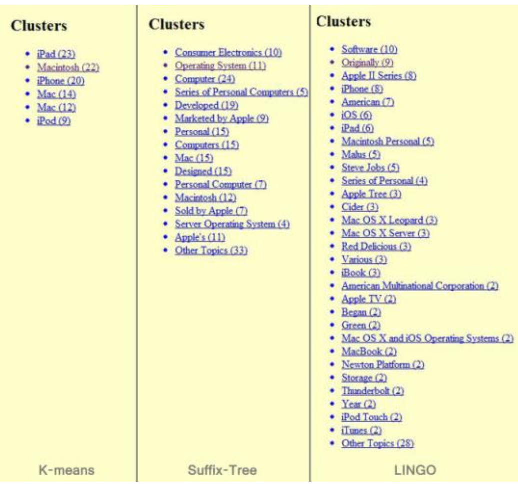
Fig. 4. Update number of search results

The search parameters can be modified from the Admin panel of the search module. The type of algorithm used i.e k-means, suffix tree, lingo can be selected. The number of search results to be fetched from the external document source can be obtained. Also the document source can be selected.

For eg. If Apple is entered as the keyword then three or more links would be generated.

1) Apple Inc.
2) Apple Fruit
3) Apple Medical Properties
4) Apple Science and Technology
5) And so on....

Each of these links consists of multiple links which is a result of clustering various sites into a single link also known as web link clustering.



Clusters

- iPad (23)
- Macintosh (22)
- iPhone (20)
- Mac (14)
- Mac (12)
- iPod (9)

K-means

Clusters

- Consumer Electronics (10)
- Operating System (11)
- Computer (24)
- Series of Personal Computers (5)
- Developed (19)
- Marketed by Apple (9)
- Personal (15)
- Computers (15)
- Mac (15)
- Designed (15)
- Personal Computer (7)
- Macintosh (12)
- Sold by Apple (7)
- Server Operating System (4)
- Apple's (11)
- Other Topics (33)

Suffix-Tree

Clusters

- Software (10)
- Originally (9)
- Apple II Series (8)
- iPhone (8)
- American (7)
- iOS (6)
- iPad (6)
- Macintosh Personal (5)
- Malus (5)
- Steve Jobs (5)
- Series of Personal (4)
- Apple Tree (3)
- Cider (3)
- Mac OS X Leopard (3)
- Mac OS X Server (3)
- Red Delicious (3)
- Various (3)
- iBook (3)
- American Multinational Corporation (2)
- Apple TV (2)
- Began (2)
- Green (2)
- Mac OS X and iOS Operating Systems (2)
- MacBook (2)
- Newton Platform (2)
- Storage (2)
- Thunderbolt (2)
- Year (2)
- iPod Touch (2)
- iTunes (2)
- Other Topics (28)

LINGO

Fig. 5. Clusters for keyword "apple"

The cluster results for the query "apple" can be seen in Fig.5. The clusters formed by using k-means, Suffix-tree & LINGO techniques are shown.

IV. Conclusion

We have demonstrated the working of dynamic link suggestion techniques which significantly improve the performance of a search engine. Clustering search results allows broadening a query to find different possibilities. There are several possible areas of future research. As mentioned earlier, if we are going to evaluate dynamic quick link algorithms for tail sites, we need to develop techniques for moving beyond track-based evaluation. We also think that there could be several improvements to our modeling, in terms of features, algorithms, and clustering. We believe our approaches, though, suggest a compelling future research direction focusing on abstracting site and link semantics. Our clustering of sites and links was heavily motivated by a hypothesis that groups of sites form cohesive classes of concepts (e.g. `restaurants', `universities'), within which there exist prototypical link classes (e.g. for the `restaurants' concept, `menu', `directions', and `reservations' links). Our results support this hypothesis, and extensions to our models should certainly be explored. The utility of these abstractions can also go beyond simple link suggestion; we can imagine a system more intelligently reasoning about a class of sites and prototypical links in response a specific user information need (e.g., `find me menus for restaurants within 3 blocks').

References

[1] J. Arguello, F. Diaz, and J.-F. Paiement, "Vertical selection in the presence of unlabeled verticals," in *SIGIR '10: Proc. of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, 2010.

[2] D. Blei and J. McAulie, "Supervised topic models," *Advances in Neural Information Processing Systems* vol. 20. 2008.

[3] A. Broder, "Taxonomy of web search," SIGIR Forum, vol. 36, 2002.

[4] D. Chakrabarti, R. Kumar, and K. Punera, "Page-level template detection via isotonic smoothing," in *Proc. of the 16th international conference on World Wide Web*, 2007.

[5] D. Chakrabarti, R. Kumar, and K. Punera, "Quicklink selection for navigational query results," in *Proc. of the 18th international conference on World wide web*, 2009.

[6] J. Seoy, F. Diazz, E. Gabrilovichz, V. Josifovskiz, B. Pangz Y, "Generalized Link Suggestions via Web Site Clustering," University of Massachusetts Amherst.

[7] O. Zamir and O. Etzioni, *Web Document Clustering: A Feasibility Demonstration*.

[8] S. Osi ́ski, J. Stefanowski, and D. Weiss, Lingo: Search Results Clustering Algorithm Based on Singular Value Decomposition.

[9] C. Carpineto, S. Osiński, G. Romano, and D. Weiss, "A survey of Web clustering engines," *ACM Computing Surveys (CSUR)*, vol. 41, issue 3, July 2009.

[10] N. O. Andrews and A. Edward, *Recent Developments in Document Clustering*, October 16, 2007.