International Journal of Computer

ISSN 2307-4531

http://gssrr.org/index.php?journal=InternationalJournalOfComputer&page=index

A Meta Search Engine Approach for Organizing Web Search Results using Ranking and Clustering

Naresh Kumar[a*], Rajender Nath[b]

[a] *Assistant Professor, MSIT, Janak Puri, New Delhi, India.*
[b] *Professor, DCSA, Kurukshetra University, Kurukshetra, Haryana, India.*
[a]*Email: narsumsaini@gmail.com*
[b]*Email: rnath2k3@gmail.com*

Abstract

The web is expanding with each passing day along with technological advancement in search engine. This results in a long list of links to be retrieved for any user query. However it is not possible to verify each link of this returned list. Even the use of page ranking algorithms in searching does not provide the desired results. To address the solution to this problem a new meta search engine is introduced that uses the similarity measurement function to determine the relevancy of web page with the given query and document clustering technique to group the results into different clusters.

Keywords: Search engine, meta search engine, relevancy, ranking, clustering.

1. Introduction:

The Web is huge in size [1] and increasing exponentially [2]. To search the information on the web a tool called search engine (SE) is used [3]. But according to [4], while searching the web, any SE can cover only 16% of the entire web. So, user cannot see those relevant pages that are not covered by SE. Moreover the study of [5, 6] showed that coverage and the precision of different SE are different and limited. So, a single SE cannot satisfy the user's choice. That's why, a user have to move towards the multiple SE to search the desired information. But it will again generate a problem of selection of SE to be used [7]. Solution to this problem is increasing search coverage and combining results of multiple SE into a single list [8]. So, a tool Meta Search Engine (MSE) is designed to address this problem [9]. MSE is an information retrieval tool that does not maintain its own index of documents and send the given query to multiple SE simultaneously [9,10]. It merges the results received from different SEs, remove the duplicate links (if available) and then present them to the user [8]. But the efficiency of a MSE depends upon the presentation of searched results and relevancy of results returned by the MSE [5], that the existing MSE are unable to return. So, this paper provides the solution to this problem by using the concept of relevancy and clustering. Relevancy refers to how much important a web page is; and clustering will group the documents according to the returned relevancy score.

---

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000 .
E-mail address: narsumsaini@gmail.com.

2. Related work

Study in [6], presents a new SE system called "Document Clustering for Search Engines (DCSE)" that was based on the concept of MSE. The main focus of DCSE was on relevancy of query and coverage of SE. Authors parse the web page for the information extraction, convert the parsed web page to tree structure and convert the document into a uniform file format. Then genetic algorithm was used to find the better solution space and to reduce the search time. Document clustering algorithm was also used to classify the results into different clusters. At the end, results showed that the proposed system reduced the amount of time required for searching the information and filtering out irrelevant data.

Oren Zamir et. al. in [11], introduced a Grouper for Husky MSE which dynamically groups the results of SE into clusters. Grouper judges the user behavior by analyzing the log files of the system. During the comparison, they consider the number of hits on each link, time spent on each link to find the user's interest and clicking distance to measure the efforts spent by the user in determining the next interesting document. They also noticed that if the number of retrieved documents increases then the number of created cluster also increased. So, they maintained it by presenting the cluster in hierarchy.

A new kind of result merging method was purposed in [10]. The MSE's query from its interface was given to three SEs – Google, Yahoo and Baidu. The proposed method used position of the words and the snippets of the page for finding the similarity between the page and the query words. From each SE and for each query the top 20 results were selected for experiment. The relevancy of returned results was also cross checked by doing manual calculation based on the defined criteria. Evaluation of results was computed based on TREC - style average precision and they found that more relevant documents appeared on the top results. The authors also concluded that it is not always true that MSE would get better result than general SE for any query.

Multi domain MSE was proposed in [12], which presents the list of domains to the user for selection. Based on the user selected domain, the list of some specialised SE was displayed. From the displayed list any number of specialised SE can be selected and any query can be given to the MSE. For performance measurement authors considered relevancy, reliability, and redundancy for extraction of data from Multi-Domain MSE. At the end, the results showed that the performance of multi domain MSE is better than the individual SE.

3. Problem Formulation:

The main problems [1,11,13,14,15], related with MSE and clustering are listed below:

a) MSE combines the results of multiple SEs together on a single platform. But currently available MSEs are not able to present the result in an effective manner.

b) Ranking algorithms working on the basis of 'Positional Ranking' and 'Count Functions' do not deal with the relevancy.

c) Number of clusters required for the results returned by the SE can be decided before downloading the web pages. But post retrieval clustering of documents can provide better results.

d) MSE like Clusty provide the results to the user in the form of named clusters i.e. it organise the links of SEs in different folders. The name of these folders is set automatically to the word which has the highest frequency in the web page. But the name assigned to these folders may not be relevant to the keywords searched by the user, as it just picks up the highest frequency word irrespective of whether they match or do not match the query keywords.

e) MSEs like Excite puts a link on the top of the ranked list if the link is retrieve by multiple SEs. But a link retrieved by multiple SEs cannot be straightaway declared as the most important link. According to [10] a link retrieved only by a single SE has been found out to be more relevant than those found by multiple SEs.

f) According to [15], 85% internet users uses internet for topic specific queries but most of them are found unsatisfied with the returned results of SE. This is because, as they does not undertake the concept of relevancy.

So the problems mentioned above can be addressed by clustering the results according to the relevancy.

4. Proposed Work

Proposed framework for MSE is shown in Figure 1, that takes into account both ranking and clustering mechanisms for organizing and presenting web pages to the user. The whole process, from giving the user query, to getting the results are organized in the following modules.

i) ***User Interface:*** This module provides the way of interaction to the proposed framework. When a user gives a query to the MSE, then this query is further provided to the multiple SEs for searching the information on the web. The returned results of the SE are stored in the local database.

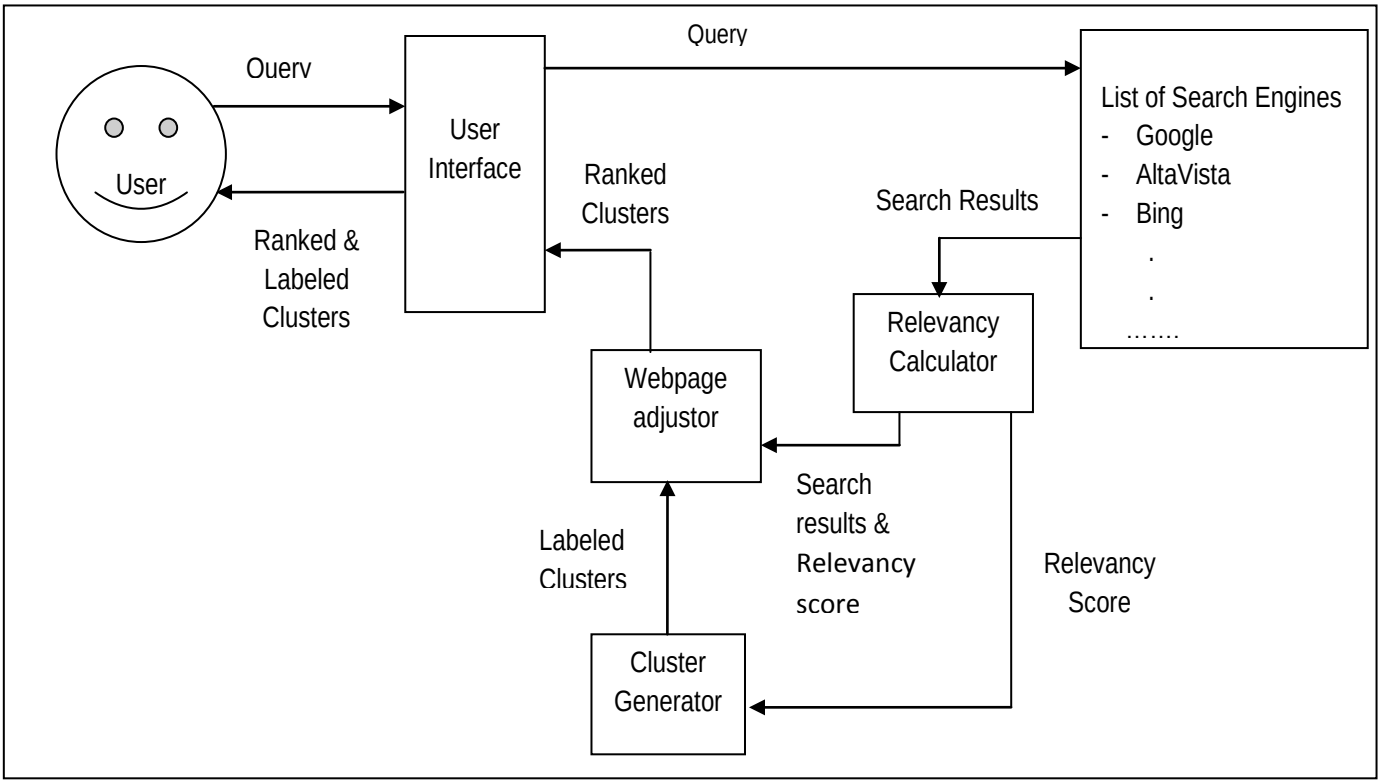

Figure 1: Proposed MSE framework

ii) ***Relevancy Calculator (RC):*** RC provides a relevancy score to each returned web page of a SE. This relevancy score is calculated for finding, to which extant the page is matching to the user query. Authors of the paper know that a relevant web page is more similar to other relevant webpages than the irrelevant pages. While there are many ways to define the relevancy based "similarity". This may include VSM, Okapi, CDR etc.. All the three named algorithms are implemented for this paper. But authors chose the Okapi similarity measure ranking formula [17] for evaluation of the proposed approach because it can perform better for both single term query search and for multiple term query search. Okapi takes two web pages as an input and correspondingly provides a similarity score. Higher score indicates better matches.

iii) ***Cluster generator (CG):*** CG is responsible for generating the required number of clusters(C). Creation of clusters is based on the lower and upper relevancy score of the web pages provided by the RC module. It also decides the relevancy range of each cluster for which URLs to be assigned. The complete process of cluster generation is illustrated by the algorithm given in Figure 2. The generated clusters are purely based on the similarity rank of the retrieved web pages.

Algorithm: Relevancy ranking and clustering
Input: User query Q, downloaded web pages (WP), number of required clusters (NC).
Output: Labeled clusters with ranked links of web pages in each cluster.
// Start of algorithm

Step 1. Get the downloaded WP of each SE separately.
Step 2. For each WP of each SE calculate the relevancy score (RS).
Step 3. [Record the lower(low) and upper(up) RS for each link of WP.
low = min(RS)
up = max(RS)
Step 4. [Generates the required no. of clusters]
R = (up – low) / NC
T = low
Step 5. [Assignment of range to clusters]

```
struct cluster
 {
        lower;
        upper;
  } C[NC];
```

Step 6. For k = 1 to NC //mark the Label of Clusters

```
{
   C[k].lower = T
   C[k].upper = T + R
   T = C[k].upper
}
```

Step 7. For each page P_i of downloaded WP with RS

```
{
  If (Pi is already visited)
   {
     Eliminate the web page and go for next iteration
   }
  Else
   {
     For each cluster Ck of NC
     {
       If (RS(Pi) >= C[k].lower) && (RS(Pi) <= C[k].upper)
        {
          Set NCk = Pi
          Exit()
        }
     }
   }
}
```

Step 8. Return the ordered clusters.
Step 9. Stop.

Figure 2. Proposed Algorithm

iv) ***Web Page Adjuster (WPA):*** WPA is responsible for removal of duplicate web pages and assignment of ranked web pages to the corresponding cluster. Organising the ranked results in the clusters is not meant to replace the traditional way of representing the search results with the new one [16]. Higher the relevancy rank of the web page then higher is the possibility for the web page to be placed on the top of

the cluster results. The complete process of assignment of webpages to the clusters is illustrated by the algorithm given in Figure 2 and a graphical representation of the same is shown in Figure 3.

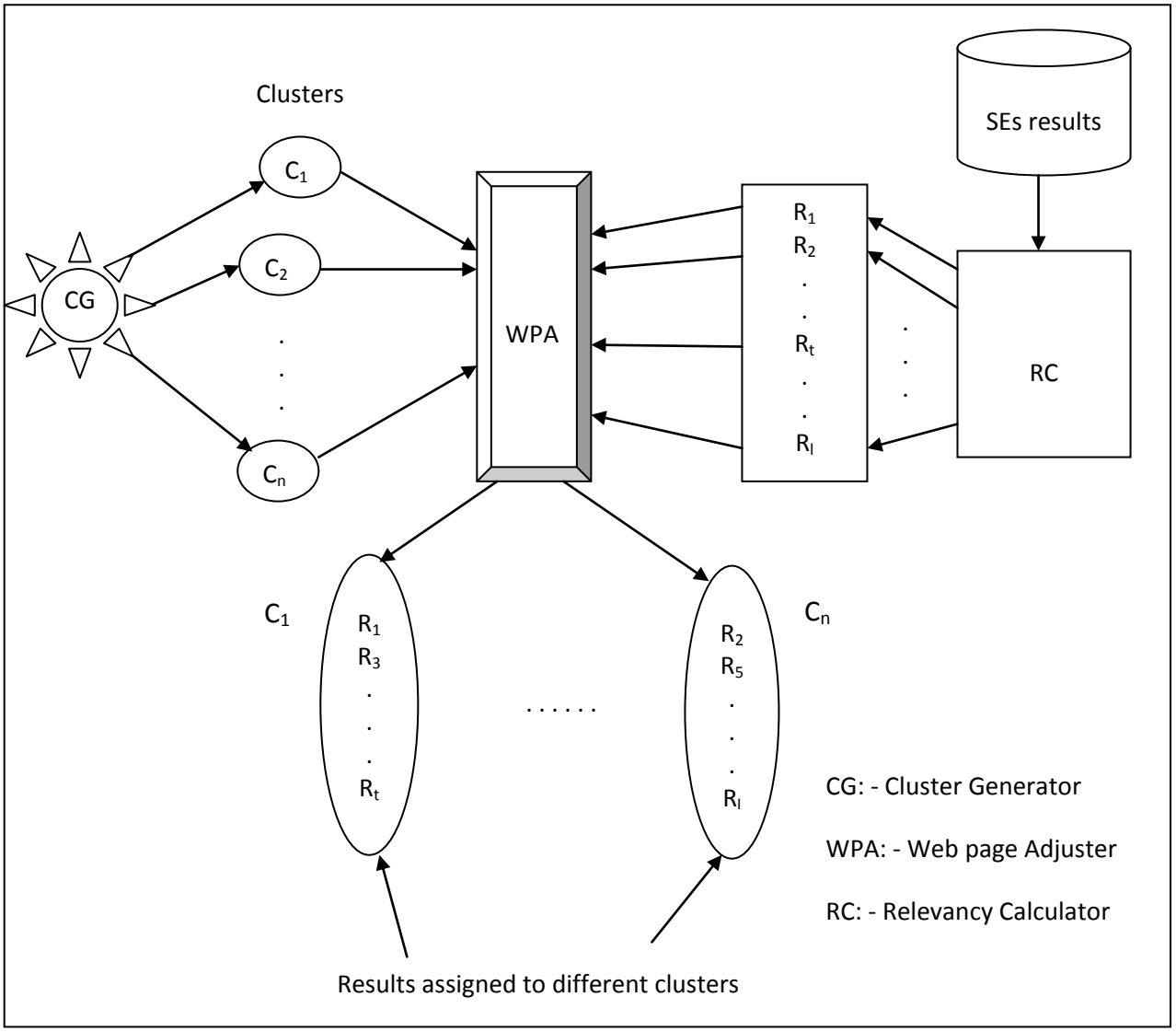

Figure 3 Clustering of MSE results

5. Experimental setup and Discussion of results:

To perform the experiment, authors chose three different SEs – Google, Bing and AltaVista.. Also, domains and queries were defined in advance as shown in Table 1. The first column in Table 1 specifies the name of domains and second column specifies the Queries available in those domains. Top 10 results are considered most relevant to the user given query. That's why top 10 links retrieved by each SE were taken for experiment. Okapi similarity formula was used to compute the similarity between the query and the retrieved results. The proposed MSE was implemented in C# programming language using the parameters defined in section 4. The proposed MSE was tested several times on 30 queries (10 queries from each domain).

The results obtained corresponding to the defined parameters showed that Most Important clusters contain 20 – 40% of the links retrieved by the SE (See Table 2).

Table 2. Percentage of Documents in Clusters

Clusters	Most Important	Partial Important	Least Important
Google	26.66	16.66	23.33
Bing	40	13.33	20
AltaVista	20	13.33	13.33

Table 2 shows the analysis of the percentage of links found in three clusters viz. Most Important, Partial Important and Least Important for each SE. For example, 40% of the links returned by Bing were clustered as most important. Hence, if a user obtained desired results from this cluster, then there will be a reduction of minimum 60% in his efforts as compared to normal search. Furthermore, 20.1% links were found redundant; So, these links were removed directly from the results before presenting them to user.

5.1 Parameters affecting the efficiency of MSE:

Furthermore; on the basis of basic concepts used, a critical look at the available MSE provides the differences (shown in Table 3) with the proposed MSE. The experimental results of the proposed work showed that ranking and clustering method returns the pages to user in organized and user friendly manner as opposed to the results returned by other MSE. According to the proposed algorithm, more similar ranked pages are grouped into the same cluster and unsimilar in other group. Now user can select any desired link of any cluster. This will reduces the search space of the user based on the range labeled with the cluster.

6. Conclusion:

This paper has proposed a Meta Search Engine to organize the results obtained from different search engines using ranking and clustering technique. The proposed meta search engine has been implemented in C# and has been tested. The experimental results have shown that the proposed meta search engine provide better results than the existing meta search engines in terms of relevancy and presentation of results. The clustering results are purely based on the similarity measure rank provided to the retrieved documents. The performance of proposed meta search engine in terms of time and space has been found a bit lesser as compared to existing meta search engines which is attributed to extra calculations done for clustering and extra code for clustering. But the relevancy achieved by the proposed meta search engine outweigh the space and time limitations.

Table 3. Comparison between different MSEs

Characteristic / Parameters	**Meta Search Engines**					
	MetaCrawler	**WebCrawler**	**Excite**	**Dogpile**	**Gnome**	**Proposed MSE**
Number of search engine used	3	2	3	3	10	3
Result Relevancy	Moderate	Moderate	High	High	Low	High
Clustering	No	No	Yes	No	No	Yes
Dynamic Clustering	No	No	No	No	No	Yes
Ranking	Eliminate delicacy and display result	Based on lexical similarity	Three-point scale	Simply collect the result and display	Page Ranking	VSM, CDR and Okapi
Re-Ranking	No	No	No	No	No	Yes

References

[1] Biraj Patel et. al., "Ranking Algorithm for Meta Search Engine", in International Journal of Advanced Engineering Research and Studies, E-ISSN2249–8974, Vol. II, Issue I, Oct.-Dec., 2012, pp. 39-40.

[2] Rajender Nath et. al. "A new Approach for Implementation of Meta Search Engine using Ranking and Clustering ", published in Satyam, MSIT journal of research, ISSN: 2319-7897 vol. 1, No. 2, Jan - June 2013, pp. 11-14.

[3] Minakov et. al.," Development of Multiagent Internet Meta-Search Engine: IT in Business (ITIB)", in International conference in St. Petersburg, June 14-17, 2005.

[4] Rajender Nath et. al., "A Novel Parallel Domain Focused Crawler for Reduction in Load on the Network" published in International Journal of Computational Engineering Research, ISSN 2250-3005, Vol. 2 Issue. 7, Nov. 2012, pp. 77-84.

[5] Yiyao Lu et. al.," Evaluation of Result Merging Strategies for Metasearch Engines", 6th international conference on web information engineering, 2005, pp. 53-66.

[6] Chun-Wei Tsai et. al., "A Document Clustering Approach for Search Engines", in IEEE International Conference on Systems, Man, and Cybernetics, October 8-11, 2006, Taipei, Taiwan, pp.1050-1055.

[7] E. Hong Han et. al.," Intelligent Metasearch engine for Knowledge Management", in Proc. of the 12th international conference on Information and knowledge management, New Orleans, LA, USA, 2003, pp. 492-495.

[8] Z.Shanfeng et. al.. "Using online relevance feedback to build effective personalized Metasearch engine", In proc of IEEE, 2nd international conference on Web information systems Engineering (WISE'01), Kyoto, Japan, 2002, Vol.1, pp. 262 - 268.

[9] Zonghuan Wu et. al.," Towards a highlyscalable and effective metasearch engine", In proc. of 10th international conference on World Wide Web, Hong Kong, 2001, pp. 386-395.

[10] Yuan Fu-yong et. al.," An Implemented Rank Merging Algorithm for Meta Search Engine", in International Conference on Research Challenges in Computer Science, pp. 191 – 193, 2009.

[11] Oren Zamir et. al.," Grouper: a dynamic clustering interface to Web search results", in Proceeding of WWW '99 Proceedings of the eighth international conference on World Wide Web, Elsevier North-Holland, Inc. New York, NY, USA, Volume 31 Issue 11-16, May 17, 1999, Pages 1361-1374.

[12] D.Minnie et. al., "Meta Search Engine with an intelligent Interface for Information Reterieval on Multiple Domains", International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol.1, No.4, pp. 37-45, 2011.

[13] D. R. Cutting et. al., Scatter/Gather: a cluster-based approach to browsing large document collections, in: Proceedings of the 15th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'92), 1992, pp 318-329.

[14] M. A. Hearst et. al., Reexamining the cluster hypothesis: Scatter/Gather on retrieval results, in: Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'96), 1996, pp 76-84.

[15] M. Kobayashi et. al.,"Information retrieval on the web," in ACM Computing Surveys, vol. 32, no. 2, pp. 144-173, 2000.

[16] Adina LIPAI,"World Wide Web Metasearch Clustering Algorithm", in Revista Informatica Economică nr.2(46), pp. 5-11, 2008.

[17] Yi Shang," Precision Evaluation of Search Engines", in World Wide Web, volume 5, issue 2, pp. 159-173, 2002.