Reranking Web Pages Based on Relevant Link from Various Search Engines

S.Amudha¹, Dr.I.Elizabeth Shanthi²

¹Ph.D Research Scholar, Department of Computer Science,
Avinashilingam Institute for Home Science and Higher Education for Women,
Coimbatore, India

²Professor, Department of Computer Science,
Avinashilingam Institute for Home Science and Higher Education for Women,
Coimbatore, India

¹amudhajaya@gmail.com, ²shanthianto@gmail.com

Abstract—In today’s electronic world search engines play a significant role in retrieving and organizing relevant information from numerous sources of the information. However, within the real ground relevancy of results made by search engines are still debatable. As a result, massive quantity of irrelevant and redundant results is retrieved. The main goal of any website is to provide relevant information to the learner. The requirement of a highly efficient and effective ranking algorithm is to provide better and relevant results based on the learner’s interest. In this paper, we have attempted a re-ranking algorithm for retrieval of relevant search engine result page in multiple search engines environment. The Re-ranking algorithm is used to change the order of the search result from multiple search engines. The result analysis shows that by using the re-ranking method based on the learner’s query, top ‘n’ web page links can be retrieved and finally learners can narrow down the search content of the web. The proposed method effectively provides information more relevant to the learners.

Keywords—Search Engine, Information Retrieval, Reranking Algorithm, Cluster, Classification.

I. INTRODUCTION

Now days everyone can be retrieve the information from the web. Internet is an essential part of our life for information Retrieval. WWW is a large repository of all type of information like text, image, audio, video, and multimedia content and it is very difficult to determine the relevant information. This information is referred by the user through search engines. Currently, there are many search engines are available to retrieve the information like Google, Yahoo, Amazon, Bing, Ask etc. The user enters the query based on retrieved the link from the search engine. The links are using re-ranking algorithm for retrieve best link provided to the user. The user passes a query, in online phase, users query is first fed up to query expansion that recognizes and eliminate ambiguity in query and pass this expanded query to search engine. Search engine gets back result on expanded query and move into the proposed re-ranking system which re-rank the retrieved result by analyzing of link of single search engine.

The users are becoming more and more dependent on the search engine ranking scheme to discover information relevant to their needs. Typically the user expects to determine the information on top ranked result and not consider document snippets. The highly ranked document is more visibility to the user and more popularity. Thus the ranking system finds the more relevant link form the multiple search engines. Currently search engines use keyword based query like single multiple and phrase searching techniques. These techniques are having some needs.

1. At the time of search process user can’t express their intention
2. Keyword matching process is difficult to identify the meaning of the keywords
3. Most matching keyword is referred as top ranked result
4. Less quality ranking of webpages based on their popularity of the web document
5. New document is not having high rank values.

In this paper analysis of the current search techniques is used based on keyword in multiple search engines. The searching results are accurately relevant to the user’s query. This system provides a high relevant to the new webpage for rank promotion. First retrieve the top N result from the various search engines based on the user query. Next step find the similarity between the links of various search engines result and assign the re-ranking values. The initial ranking value is based on find the important score for the links. Finally combine the important score and similarity score get new ranking values.

The remainder of the paper is organized as follows: The next section 2 presents related works. In Section 3 Research methodology is described where we explained in 3.1 pre-processing techniques, hierarchical clustering in 3.2, SVM classification in 3.3 and lastly used them Re-ranking algorithm in 3.4. Section 4 includes experimental evaluation and Result estimated. The remaining, Section 5 will includes conclude the paper by giving a brief of the main contributions of our approach and presenting future directions.

II. RELATED WORK

Anuradha R Kale et al [1] has developed a re-ranking system based on the user profile information. This system proposes a novel re-ranking algorithm works semantic similarity to improve the quality of SERP of website. The first step fetch the top N result from the search engine and match...
the similarity between the candidate and query based re-ranking result. Then calculate the importance score based on the rank position in the every candidate and after combine the similarity score and the importance score to produce a novel re-ranking algorithm. Finally produce the better result compared with the existing algorithm.

Falah H et al [2] has developed ranking web pages using collective knowledge. This system assigns the index value to the web document using hybrid techniques. This experiment used TREC datasets provide both indexing and retrieved relevant web pages. This model provides a variety of analytic capabilities, including: concept extraction, concept correlation, text summarization, spam filtering, and term to document similarity.

Gunhan Park et al [3] has proposed a re-ranking algorithm using post-retrieval clustering for content-based image retrieval (CBIR). In this conventional CBIR system, utilize the similarity relationship of the retrieved results via post-retrieval clustering. The first step images are retrieved using visual features such as colour histogram and the retrieved images are analysed using hierarchical agglomerative clustering methods (HACM). Finally assign the rank of the results is adjusted according to the distance of a cluster from a query. Furthermore, this system analyse the effects of clustering methods, query cluster similarity functions, and weighting factors in the proposed method. This system compared a number of experiments using several clustering methods and cluster parameters. Experimental results show that the proposed method achieves an improvement of retrieval effectiveness of over 10% on average in the average normalized modified retrieval rank (ANMRR) measure.

Harish Kumar B T et al [4] has developed a new method to rank to the relevant pages base on the content and keyword. Every result page contains page rank values provided by the search engine based on the user query. The user query is preprocessed for identify the root words. The root word compared with the dictionary and matched is founded then assigns the weightage value. Finally the total relevancy is computed and re-ranked results order in descending order.

Jiafeng Guo et al [5] have proposed a new Re-ranking algorithm for Named-page for PSM. This system finds the named page and requires the target document as rank first. The new ranking algorithm PSM is improves the performance and effectively reorder the webpage. Finally provide the better performance compared with the traditional methods.

Ms.S.Muthukakshmi et al [6] has developed a new method to efficiently provide better Web page recommendation. This web pages recommendation through semantic enhancement with integrating the Web usage knowledge and domain. The web page candidates produce a number of effective queries. Web page Recommendation systems can take advantage of semantic network to overcome common limitations of current systems and improve the recommendations’ quality.

Pengjie Ren et al [7] has proposed a novel User Session Level Diversification approach based on the observation that a query’s subtopics are implicitly reflected by the search intents in different user sessions. This system consists of two phases are Session Graph Construction and Diversity Reranking. For a given query, Session Graph Construction builds a Session Graph which considers relevant user sessions and preliminary retrieval results as nodes and the nodes’ pairwise similarities as edge weights. Diversity Reranking to assign the re-ranks the preliminary retrieval results by minimizing a Session Graph based diversity loss function. Extensive experiments on two standard datasets of NACISIS Test Collections for IR (NTCIR) demonstrate the effectiveness of our approach. The advantage of our approach lies in its ability of avoiding mining the query subtopics in advance while achieving almost the same or better performances compared with previous approaches.

Renjini L et al [8] have proposed a personalized search engine with re-ranking algorithm. This experiment creates a new page rank algorithm and re-ranking algorithm. This system using query expansion consists of term frequency of keyword searched by the user log. Re-ranking algorithm is reorder the search result according to URL, click history and identify last action performed by the user. Finally, the result analysis shows that by using the re-ranking method based on user activities, the precision, recall and NDCG values for search results show significant improvement.

Shital C. Patil et al [9] has proposed work a new approach is introduced to re-order the search results based on the contents and user interest rather than keyword and page ranking provided by search engines. Based on the user query, search engine results are retrieved. Every result is individually analyzed based on the user query and page contents and particular score is awarded to each result. Finally, the relevancy of the particular link against user request is computed by summarizing all the scores and the reordered list is displayed to the user. When the user visits the web page out of this reordered list, the query, url and the contents extracted from the web page are stored in the server log. When next time user enters a query the scores are awarded to each result link based on the data in the server log which indirectly incurs the user interest.

Ganesh venkataraman et al [3] has developed a method of re-ranking the search results that have been primarily ranked using either conventional algorithms that use link structure and user clicks or semantic algorithms, using a combination of general webpage features and user interests. The features of web pages like images, videos etc., are extracted by crawling them and the user's general interest in those features are learnt from past queries made and clicks on particular results. Using the degree to which each feature is present and the corresponding interest of the user, the user's interest in a particular search result is predicted and consequently the results are re-ranked in such a way that it augments the efficiency and effectiveness of conventional intent / meaning driven semantic search concept.

III. RESEARCH METHODOLOGY
The figure 1 shows the framework diagram for the proposed system and its retrieve the relevant link from the various search engines.

A. Pre-Processing:

Pre-processing technique plays a very important role in the text mining techniques and applications. Pre-processing is a first step of text mining process. Figure 2 shows the steps in pre-processing for retrieved SERP in various search engines are processed like tokenization, stop word removal and stemming.

The figures 3 represent the tokenization of all the web page content, operator separator and transfer to the next step Stop word removal. The figure 4 shows how to remove the stop words like of, the, then, that, we, are, as, at etc. The figures 5 represent the stemming process to find the exact keywords.

B. Hierarchical Clustering

Hierarchical clustering algorithm is a given set of N items to be clustered and also find the similarity matrix N*N distance. The following step for process the group of similar content in webpages link in various search engines like Google, Yahoo, Bing, and Ask.

1. Every item is assigned to the cluster as N items and N cluster. Then find the similarity between the clusters the same as the similarity the items.
2. Next find the most similarity content link to form a pair of cluster is called Single cluster.

3. Calculate the similarities between the new cluster and the existing cluster.
4. Repeat step 2 and 3 until reach single cluster of size N.

C. SVM Classification

Support Vector Machine is a supervised machine learning algorithm for classification and regression. SVM is finding a hyper plane for the webpages content and link data.

The optimal hyperplane have specified an infinite number of various methods used by scaling of the variable $\beta$ and $\beta_0$. The hyperplane formula shows as $|\beta_0 + \beta^T x| = 1$.

The x symbol represents the training data near to the hyperplane is called support vectors. This also called as canonical hyperplane. The distance between a point x and hyperplane $(\beta, \beta_0)$ is produce the result of geometry.

$$\text{Distance} = \frac{|\beta_0 + \beta^T x|}{||\beta||}$$

The distance support vector is known as the numerator is equal to 1 and the distance to the support vectors.

$$\text{Distance support vectors} = \frac{|\beta_0 + \beta^T x|}{||\beta||} = \frac{1}{||\beta||}$$

The margin $M$ is twice the distance to the closest.

$$M = \frac{1}{||\beta||}$$

Then based on the constraints the problem of maximizing $M$ is equal to the problem of minimizing function $L(\beta)$. The constraints model is based on the requirement for the hyperplane to classify correctly all overall training data $x_i$ and $y_i$ shows each label of the training data.

$$\min L(\beta) = \frac{1}{2} ||\beta||^2 \text{subject to } y_i (\beta^T x_i + \beta_0) \geq 1 \forall i, \beta, \beta_0$$

D. Re-Ranking Algorithm

The reranking algorithm based on the relevancy of the keyword and content in the webpage. Reranking algorithm is giving higher weight to document having most relevancy measure. Our novel algorithm should be ranked higher for content and topic related to the user query.

Algorithm: Relevancy based approach

Input: User Query (OQ)

Output: Re-Ranked URLs

Variables List

OQ : Original User Query
PQ : Pre-processed User Query
URL_LIST[N] : Array Storing all the URLs Returned by Google, Yahoo, Bing, and Ask API
RW[N] : Array for storing important words of query PQ
KW[N] : Array for storing key words from the URL
CW[N]: Array for storing Content words from the URL
URLi_BODY
URL_RERANK[N]: Array to store scores computed for each URLi
Steps:
URL_LIST[N] <- GOOGLE_API(OQ), YAHOO_API(OQ), BING_API(OQ), ASK_API(OQ) /* Retrieve link from multiple search engines PQ
Preprocess(OQ);
//* Preprocess the original query
RW[N] <- Extract Root Word From Query PQ /* Extract the root word from the preprocess query
For each RWi in RW[N] // match root word in the list of words in link
Find the Synonym Si for RWi
Add Si and RWi to Dictionary D
End for RWi
For each URLi in URL_LIST[N] // preprocess the url, content
URLi <- Preprocess(URLi);
URLi_KW[N] <- Extract_Keywords_From(URLi)
URLi_BODY <- Preprocess(Body tag text of URLi)
URLi_CW[N] <- Extract_Content_Words_From(URLi_BODY)
End for URLi
For each URLi in URL_LIST[N] // compare the keyword and url content assign ranking value
For each KWi in URLi_KW[N]
Compare URLi_KW[n] URL_LIST[N]
If match then
URLi_RERANK <- URLi_RERANK + 1;
Else
URLi_RERANK <- URLi_RERANK + 1;
End for KWi
For each CWi in URLi_CW[N]
Compare URLi_CW[N] URL_LIST[N]
If match then
URLi_RERANK <- URLi_RERANK + 1;
Else
URLi_RERANK <- URLi_RERANK + 1;
End for CWi
End For URLi
Sort URL_LIST[N] based on URL_RERANK [N] in descending order

IV. EXPERIMENTAL EVALUATION AND RESULT DISCUSSION
The user enters the query as ‘nodejs’ to pass the various search engines and retrieve the webpages as based on the window size. Table 1 shows the result of the user query ‘node js’ and retrieve the webpages, rank values of the current search engine and the window size. The table 2 shows the comparison of ranking values of various search engines for find the similar links. The figure 6 shows the comparison of link and ranking values from search Engines: Google, Yahoo, Bing, Ask. The Table 3 shows the extracting relevant link and the figure 6 shows the reranking of Extracting Relevant Link of various search engines.
Figure 4: Stop Word Removal

Figure 5: Stemming

### Table 1: Extracting Link from Search Engines: Google, Yahoo, Bing, Ask

<table>
<thead>
<tr>
<th>Searching Link</th>
<th>Search Engine Name</th>
<th>Rank</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>node js//ask//https___eloquentjavascript.net_20_node.txt</td>
<td>ask</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>node js//ask//https___en.wikipedia.org_wiki_Node.js.txt</td>
<td>ask</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>node js//ask//https___github.com_nodejs.txt</td>
<td>ask</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>node js//ask//https___github.com_nodejs_node.txt</td>
<td>ask</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>node js//ask//https___nodejs.org_.txt</td>
<td>ask</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>node js//ask//https___nodejs.org_en_download_.txt</td>
<td>ask</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>node js//ask//https___twitter.com_nodejs_lang=en.txt</td>
<td>ask</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>node js//ask//https___<a href="http://www.w3schools.com_nodejs_nodejs_intro.asp.txt">www.w3schools.com_nodejs_nodejs_intro.asp.txt</a></td>
<td>ask</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>node js//bing//https___docs.npmjs.com_files_package.json.txt</td>
<td>bing</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>node js//bing//https___github.com_nodejs_node_blob_master_doc_changeslogs_CHANGELOG_G_V8.md.txt</td>
<td>bing</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>node js//bing//https___nodejs.org_en_.txt</td>
<td>bing</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Link</td>
<td>Google</td>
<td>Yahoo</td>
<td>Bing</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>--------</td>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td><a href="http://iconof.com/blog/how-to-install-setup-node-js-on-amazon-aws-ec2-complete-guide/">http://iconof.com/blog/how-to-install-setup-node-js-on-amazon-aws-ec2-complete-guide/</a></td>
<td>0</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td><a href="http://node.green/">http://node.green/</a></td>
<td>0</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td><a href="http://www.oreilly.com/ofps/">http://www.oreilly.com/ofps/</a></td>
<td>0</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td><a href="http://www.softpedia.com/get/Internet/Servers/WEB-Servers/Node.shtml">http://www.softpedia.com/get/Internet/Servers/WEB-Servers/Node.shtml</a></td>
<td>0</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td><a href="http://www.tutorialspoint.com/nodejs/">http://www.tutorialspoint.com/nodejs/</a></td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td><a href="https://docs.microsoft.com/en-us/javascript/azure/?view=azure-node-latest">https://docs.microsoft.com/en-us/javascript/azure/?view=azure-node-latest</a></td>
<td>0</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td><a href="https://docs.npmjs.com/files/package.json">https://docs.npmjs.com/files/package.json</a></td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td><a href="https://eloquentjavascript.net/20_node.html">https://eloquentjavascript.net/20_node.html</a></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><a href="https://en.wikipedia.org/wiki/Node.js">https://en.wikipedia.org/wiki/Node.js</a></td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**TABLE 2: COMPARING RANKING VALUES FROM SEARCH ENGINES: GOOGLE, YAHOO, BING, ASK**
Table 3: Extracting Relevant Link

<table>
<thead>
<tr>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://www.tutorialspoint.com/nodejs/">http://www.tutorialspoint.com/nodejs/</a></td>
</tr>
<tr>
<td><a href="https://docs.npmjs.com/files/package.json">https://docs.npmjs.com/files/package.json</a></td>
</tr>
</tbody>
</table>
https://en.wikipedia.org/wiki/Node.js
https://en.wikipedia.org/wiki/Nodejs
https://nodejs.org/
https://nodejs.org/en/
https://nodejs.org/en/download/
https://www.w3schools.com/nodejs/
https://www.w3schools.com/nodejs/default.asp
https://www.w3schools.com/nodejs/nodejs_intro.asp

V. CONCLUSION AND FUTURE WORK

Reranking method is implemented in various search engines to improve the quality of searching results. This system fetches top N result by using the similarities between webpage content, metatag and title of the link based on the query. The extracted webpages are preprocessed using tokenization, stopwords removal, stemming. The next step clustering process group the similar of webpage link and classify the web page content, link and metatag. Finally assign the reranking values to the fetch top similar result of various search engines. The experimental result shows that more relevant web pages are displayed to the user using novel re-
ranking algorithms. Our future work is to improve the reranking model based on the user behavior able to query auto completion, and to effectively capture the relationship between user behavior and retrieval of web pages in various search engines.

REFERENCES


