Optimized Web Search Results Through Additional Retrieval Lists Inferred Using WordNet Similarity Measure

Saravanakumar K[1], Aswani Kumar Cherukuri[2]

School of Information Technology and Engineering,
Vellore Institute of Technology University,
Vellore, India
[1]ksaravanakumar@vit.ac.in
[2]cherukuri@acm.org

***Abstract*— Search engines have become mandatory part in the usability of information available through Internet. They provide direct support in the growth of the World Wide Web. Today their concern is to give more importance to the precision in the top results suggested to reduce the iterative search of a concept by any user. Our main objective is to improve the efficiency of search results suggested by the search engine in response to a query. The objective is approached by constructing alternate queries for the main query given by the user. It involves the selection of contextually most similar alternate queries through the method proposed here. The coalition of results produced by the main query and the alternate queries could improve the precision in the top pages. We evaluated the proposed method and observed that the proposed method performed well in projection of some of the important links of the search results into the top few pages. Also, it is observed that the precision improved considerably.**

***Keywords*— Information Retrieval; Semantic Relatedness; Alternate Query Formation; Re-ranking Search Results; Wordnet Similarity.**

I. INTRODUCTION

The most important purpose of the internet today has become finding or gathering the information that we need. In most of the search engines, the search is based on the keywords present in our query that we submit [13]. Search engines take those keywords and search the pages for the presence of those words and produce the result as set of web pages. As the result set contains millions of suggestions, no one can explore all suggested links to end up in what they need. Here comes the role of ranking of suggestions by the search engines. The ranking of the pages are mostly done by popularity or by the number of hits the page has, but not on the relevancy of the information user wants to retrieve, in most cases. Some popular methods which would be treated as good platform for Information Retrieval and related domains are discussed by Baesa-Yates in their work [15]. While we are processing a given query, the semantic meaning or synonyms of keywords or sometimes the colloquially used synonyms of a given word must be considered. The use of such knowledge or sources would improve the search result considerably. This must be done in addition to the conventional methods. Failing which may lead to irrelevant suggestions or ranking of the pages. Ontology could be used as a component of knowledge source in the process of understanding the query [14]. The re-ranking problem is given importance in recent times to attain high precision at the very top ranks. In this process, the components like Ontology, thesaurus and dictionaries are used to better understand the query [14, 16].

For a query, "popular tourist spots of India", any search process would explore and retrieve pages matching the keywords popular, tourist, spots and India in most cases. Conceptually the query could also mean "famous tourist spots of India", "Top tourist destinations of India", or "Famous travel destinations of India". These queries show that even though most of the words are direct synonyms of the keywords present in the given query, still some of them are not direct synonyms. Even the words might not be part of Thesauruses sometime. Though these are matching well with the query given in the question, most of the search engines do not consider the pages which include these alternate key words. Hence it is always suggested to take into consideration the semantic meaning and the synonyms and also the colloquially used relevant words for searching. This method of finding alternate queries using various resources sometimes considered in literature as paraphrasing which is applied in variety of natural language processing applications like question answering system, text categorization, text summarization, machine translation, etc [8].

The approach we are proposing here contains the following steps: we pre-process the query in order to remove the stop-words. Only the keywords which are the candidate words in deciding the semantics of the sentences will be taken into consideration [5]. Then, we construct a vocabulary for those candidate words using their synonymies using WordNet [6]. We have found that other semantic and lexical meaning finding ontology based dictionaries are providing good knowledge but they sometimes lack efficiency and precision. WordNet is designed to be used as dictionary or thesaurus and to support text analysis. Hence, we used WordNet as lexical source for solving our problem. Finally, we construct the alternate queries with the use of candidate words of the given query, analyze the results of those queries, and accordingly re-rank the final result of the given query.

The rest of this paper is organized as follows; section 2 discusses about the related work, section 3 discusses about the

978-1-4799-4674-7/14/$31.00 ©2014 IEEE

proposed architecture we used here to improve the performance of the search system and provides the clear description about the methodology used, and section 4 discusses about the experiments and results supporting our work, finally with conclusion and future work in section 5.

II. Related Work

It is apparently observed that the keyword based search would not fetch the relevant results. The modern search engines heading towards lexical knowledge bases. Hence, searching for documents using any queries along with their semantics equivalents would improve the search efficiency. Mustafa et. al. proposed a similar work to find the semantic meaning of the search keywords given by the users to make the search results more relevant [9]. The short requests in the form of query causes a problem called "Ranked list problem" of Web searches [7]. Sometimes, polysemous word tends to be ambiguous in a sentence which in turn increases the number of attempts by any user to obtain the relevant links.

Query expansion is the term used with expanding a query based on different criterions. For example, the query can be expanded with the help of ontology, thesauri, interactive queries, relevance feedback etc [15]. The queries are much sought to expand because of the fact that most of the users do not know their domain of search terms availability. Reformulation of a query is one of the techniques to expand a given query [1].

The main function of any language is to pass along some information. Hence the syntax of any language is to mold the proper organization of words into sentences. Tagging a word with proper Parts Of Speech (POS) tags in a given sentence would help much while reformulating the query [11, 18, 24]. As there is more than a way of delivering a concept, there is more than one syntactic placement which could deliver the same message. Here, parts of speech blocks used to capture syntactic arrangement in language. The assumption used here is that the most frequently happening POS blocks in language must be the ones that gain most information content possible in the least effort-consuming way. The sentence classification approach in [18], assign exact class label to the sentences according to their senses or their meanings. In this approach the specificity scores are calculated and by the classification done with the group of sentences can be used in reviews or for product survey reports. Feritas et. al., have also used the concepts of semantic relatedness in their work [24]. The idea of semantic similarity, they have used to find the similarity between two words of the user query over the linked data of Wikipedia pages. Among various links from a concept's page, they identify the relevant one using the semantic similarity. But they used this concept only in Wikipedia data and the links inside Wikipedia. They have not applied this over WWW for Information Retrieval purpose. Their work mainly related to Question Answering using semantic relatedness over linked data.

Many approaches have been proposed on improving search efficiency are also based on Ontology, and Formal Concept Analysis. Ch. Aswani Kumar et. al., have investigated the effectiveness of Vector Space model, Formal Concept Analysis, and Latent Semantic Indexing in Information Retrieval [25]. The analysis clearly shows the application of those concepts in information Retrieval leads to a better ranking of documents. Ontology, in other words would signify the specification of a conceptualization in the domain of knowledge sharing. More generally, it is a description of the concepts and their relationships that can exist for a factor or a group of factors. It is used to understand various entities within that domain, and could also be used to distinguish the domain. The ontology based information retrieval mechanism has been proposed and widely used to improve the precision of the search results [9, 17]. In all the cases, we found that re-ranking of the search result is done to improve the precision at very top ranks based on inter-documental similarities or co-occurrence of terms (keywords) presents [12, 17]. The main aspects found through related works are given precisely in table 1.

TABLE I
Related Literature

SNo	Authors	Observations
1	Gloria et al. (2012)	Proposed an iterative query disambiguation mechanism to optimize web searches of a short query and to eliminate "ranked list problem"
2	Lior et al. (2011)	Proposed a method to improve precision at top ranks through comparison of search results of a query using two different search algorithms. Further exploits inter-document-similarities between the lists produced.
3	Saravanakumar & Deepa, (2011)	Expansion of the query using WordNet and to improve the search results through iterative searching.
4	Sathya et al. (2011)	Proposed Intelligent Cluster Search Engine to improve the search efficiency by involving clustering of documents through comparison of co-occurrence terms.
5	Shailesh et al. (2010)	Proposed an approach to improve the search by identifying the specific sentence in a given text using specificity score. Further used, POS to develop a classifier to classify the specific sentences.
6	Lioma & Ounis, (2008);	Proposed syntactically based query reformulation technique using shallow syntactic structures (POS blocks)
7	Angel et al. (2005)	Query expansion using similarity thesauri. To improve short queries by constructing thesaurus of similarity between terms.

Similar work has been suggested to construct alternate queries through finding the semantic meaning of the search keywords [16]. Using the exact keywords often fail to retrieve the information sought by the user. Hence, using WordNet, the semantic meanings of the words are found then they are

combined to make the alternate queries. We would use these alternate queries to search for target documents. Here, we proposed a technique to choose the most similar alternate queries to the given query and context.

III. Proposed Architecture

The goal is to make the top results more relevant to the user query as well as cover all the possible semantic meanings by which the user will not have to search with different keywords to get the needed information. The proposed architecture is shown in Figure 1. The steps are as follows;

A. Query Pre-processing

The Query Preprocessing is essential and considered to be indispensable to refine the given query and eliminates the unnecessary words to make the search process easy and reach the target document which the user actually intended [4]. The "Stop-words" (Ex: a, an, the, in, on etc) will be removed as their contribution in identifying the target documents are minimal and it is well known through many literature. Most of the search engines remove the stop-words while searching, so these words are removed to make the query refined in a manner where we can apply retrieval algorithms effectively as discussed in [3]. After the removal of "Stop-words", we check for the spelling of each keyword in the query. If there is any mistake made by the user it is corrected and the query is ready to be processed by our proposed methodology.

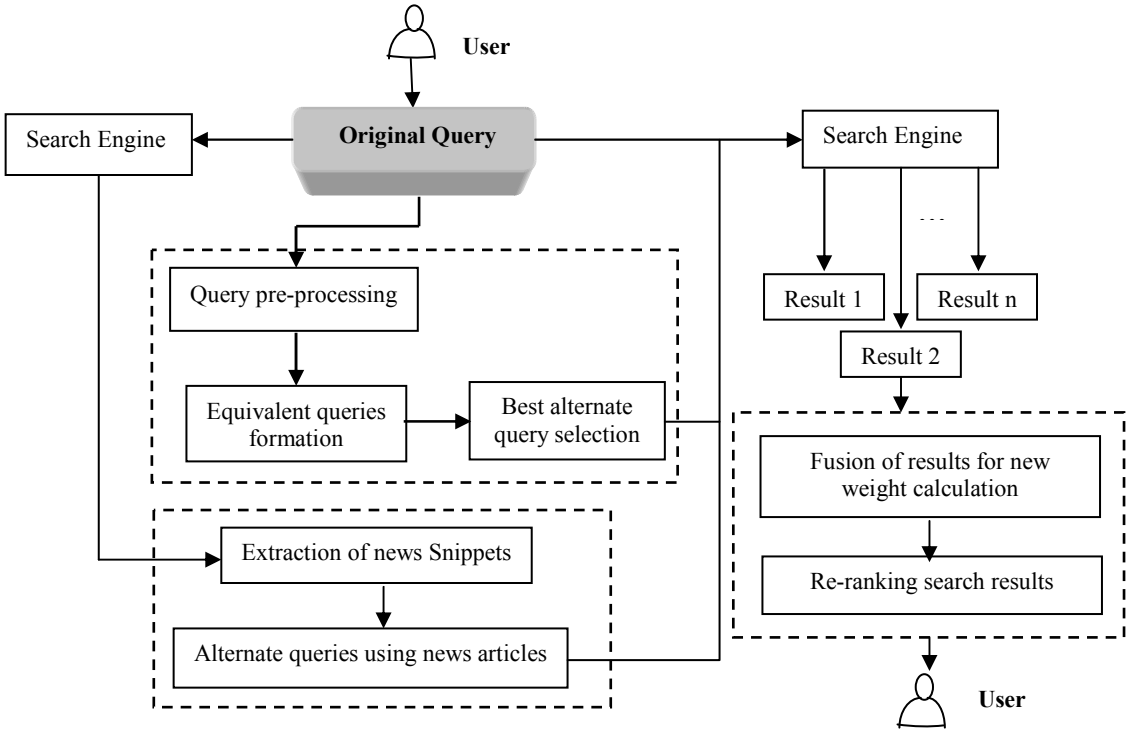

Fig 1. Architecture of the proposed system

B. Alternate Query Identification

In this step the similarly meaning alternate queries will be formed and most relevant among them with the base query are identified. First we look for the semantic meanings of each keyword. Each keyword will have different semantic synonyms which we get using WordNet [21]. After fetching the synonyms we combine the synonyms with one another. To simplify the analysis, in the query "Famous tourist spots", the word "Famous" would mean celebrated, famed, notable etc. as synonyms, the second keyword "Tourist" means tourer, holidaymaker etc. and the third keyword "spot" has synonyms like destination, place etc. Now we form the alternate queries using different combinations of all the synonyms of keywords, in the same order as the main query, we get queries like: "Famed tourist spots", "famous travel destinations", "famous tourist places" etc. Now what we got is a set of alternate queries. The investigation on search result improvement using this set could project some of the documents to the top ranks as suggested in [16]. But the problem with all these queries which formed through this method is that not all of them are relevant. Paraphrasing using the synonyms of keywords does not necessarily mean that all of them are relevant at least for a given context. The reason is that although English language has many synonyms such as 'large' or 'big' for a word 'large', it is unusual for these words to have exactly the same meaning in different contexts. So, while we paraphrase using different vocabulary, we might end up in sentences that are not natural English. Contrary sentences are likely to reduce meaning of the whole sentence more as because one may not understand what we try to say. Hence, to avoid this problem, we find the "Relatedness Score" (weights) between each set of keywords present in the query for the calculation of the total score [10] as shown in Figure 2.

In our proposed method, we used Gloss Vector measure for finding the similarity score between two words. The Gloss Vector measure (vector) works by making second-order co-occurrence vectors from the WordNet definitions of concepts. According to the experimental results of Pedersen et al., the Gloss Vector measure performs favorably well to other measures in respect of semantic relatedness, and performs comparatively well in word sense disambiguation [19, 20]. Besides, the proposed measure can also be used to compare any two concepts or glosses without considering their parts of speech. In addition, the way of measuring the relatedness is adaptable since any corpora can be used to derive the word vectors. We further divide the result into two categories: Relevant and Non-relevant. We take the relevant queries into consideration and discard the Non-relevant one. Example of calculation of weights is depicted in Figure 2. We have taken the search query "American basketball team" to discuss the example. For the query "American basketball team", the sentence *"American#n#1 basketball#n#1 team#n#1"* shows the maximum weight through gloss vector measure, because of the similarity values of the keywords' senses. The similarity score we found this way is used to calculate the total score made by each query. We used the top valued set of alternate sentences as threshold value for further processing. At last, the alternate queries are submitted to the search engine and the results are stored.

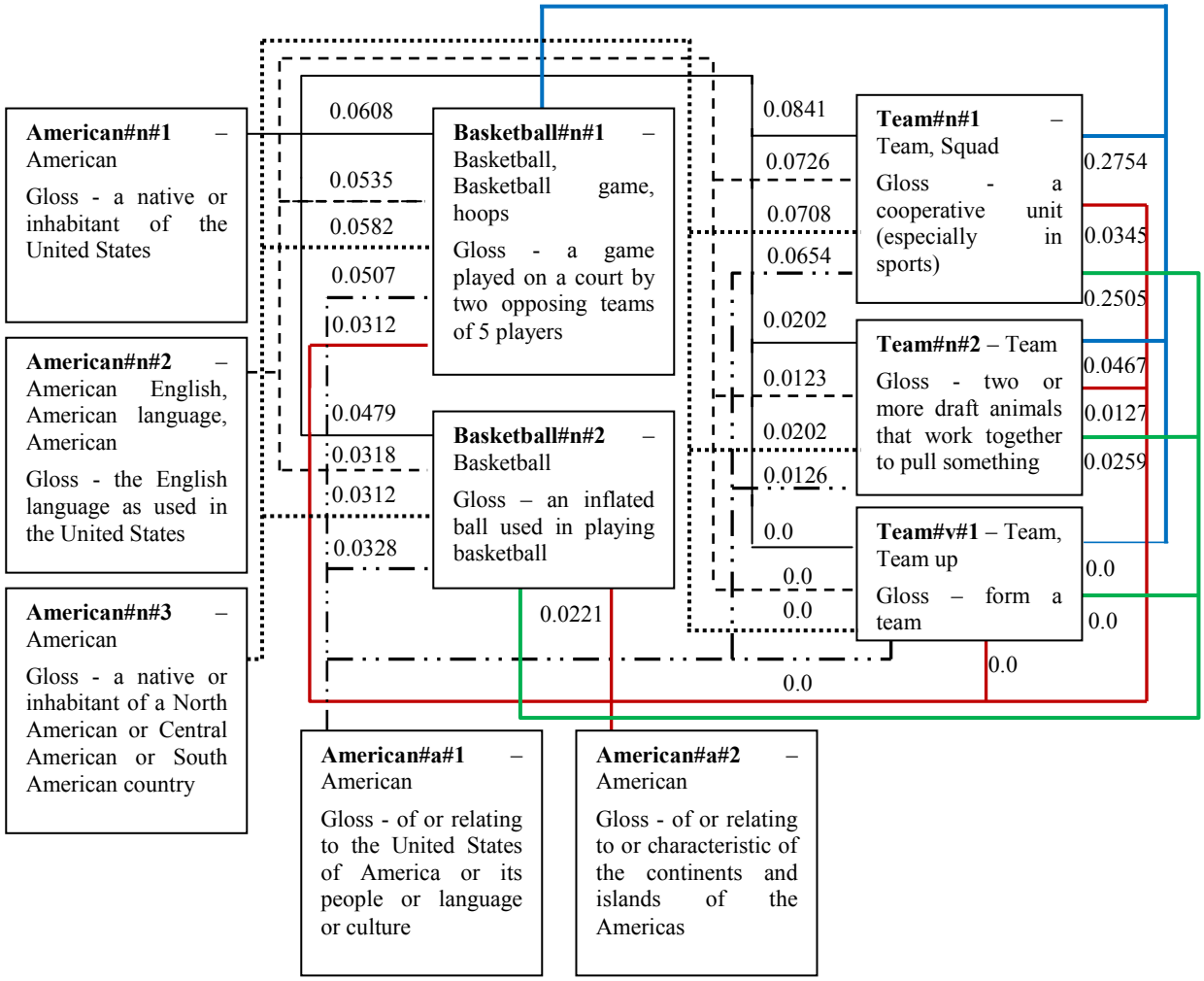

Fig 2. Weight calculation for the query "American basketball team"

C. Equivalent Query Formation

After identifying the alternate queries we look forward to form the Equivalent Queries, that is, the set of queries which could be formed using the highly correlated meaning in different contexts. It has been observed that the dictionary synonyms are not always enough to get all the meanings of a word as users sometimes use the "Colloquial Terms" for search, instead of actual word. Hence, the Colloquial Terms are considered as important inputs in our approach. For example, the query mentioned in the Introduction section, "popular tourist spots of India", would mean "famous tourist spots of India", "Top tourist destinations of India", or "Famous travel destinations of India". When carefully look

into this equivalent query list of the given query, the words famous, and top are not the synonyms of the word popular. But we do use them interchangeably because of their relevance in contexts. In order to find such terms, we could use Blogs, or News sites. Here, we consider the search result snippets' titles of the first page of the search result given by the search engines to identify the set of possible alternate queries which could be derived for the given one. Just like the previous step we measure the Relatedness Score for the terms found here and according to the scores we identify the relevant equivalent queries.

D. Re-ranking of Search Results

Initially we have 3 result sets: 1. Results produced in response to the Original Query by the search engine (Result set M), 2. Results produced in response to the set of Alternate Queries (Result set A_i) and 3. Results produced in response to the set of Equivalent Queries (Result set E_n). All the links suggested in each result set have their own "Rank" which is given by the search engine. It is known that the results are arranged according to their weight or rank in descending order. We assign each page a "Weight" which is allocated by their rank as search results. Hence all the pages have some score according to their rank. The proposed system algorithm is depicted in Figure 3.

Algorithm 1 *CALCULATE_WEIGHT*(L_i)

INPUT: Main result set (M), Alternate queries result sets (A_i), Equivalent queries result sets (Ei)

OUTPUT: Main result set in new order (M_{new})

1: **for all** URLs $L_i \in$ M **do**
2: $W_i \leftarrow$ Rank as weight
3: **for each** URL K_j of every $A_n \in A$ **do**
4: **if** L_i presents in K_j **then**
5: $W_i \leftarrow W_i$ + (Rank of URL K_j / Number of links in current result set A_n)
6: **end if**
7: **end for**
8: $W_{new} \leftarrow W_i$
9: **for each** URL R_j of every $E_n \in E$ **do**
10: **if** L_i presents in R_j **then**
11: $W_{new} \leftarrow W_{new}$ + (Rank of URL R_j / Number of links in current result set of E_n)
12: **end if**
13: **end for**
14: **return** W_{new}
15: **end for**

Fig 3. Proposed system algorithm

The weight to each URL in the main query search result is calculated as follows; if an URL appears in all the three result sets it will be allocated more weight than the one appearing in one result set. Let us suppose that for given query Q in search engine we get 100 resultant URLs which is the main result set M. For the link suggested as first link in M, we assign the weight 10; second link will be assigned 9.9; third will be 9.8 and so on. If the URL of M appears in another result set of alternate queries or equivalent queries, then a weight is added to the already assigned weight based on its position in the alternate queries result set. This is done for each URL, and finally all the URLs of M will have a new weight score. The equation follows used to calculate the new weight of a given URL.

$$W_{new} = W_{main} + \sum_{i=0}^{n-1} (L_p * W_i)$$

Where W_{main} is the weight assigned initially based on the position in the main result set M, L_p is Boolean value 0 or 1 based on the presence of an URL in any of the target result sets, W_i is the weight added to the URL if it presents in the target link and it is calculated using the position, and W_{new} is final weight of an URL. Finally the re-ranked results are given to the user in response to the query submitted to the search engine. The result set will contain the re-ranked URLs based on the new weights.

IV. Results and Discussions

We used WordNet 3.0 [6] for the English language to derive synonyms and to find semantic similarity between words in calculating the weight of each alternate query derived. We have studied the performance of the proposed system with around 200 queries. We have tabulated some of the queries in Table 3 along with the Precision for Google results and the proposed system. It has been widely acknowledged that the users are not interested to go beyond two or three pages [2]. Hence, the precision is calculated as precision at a given cut-off rank using the following equation.

TABLE IIII
Experimental Results

Query	Count of alternate queries	Count of matching queries	Count of URLs projected to top 20 from later positions	No. of matching URLs in projected links	Precision (%)	
					Google	Proposed method
American basketball team	19	6	14	9	70	75
Algorithm to find prime numbers	602	24	9	5	55	80
Effect of global warming	102	8	10	7	70	85
Baseball bat manufacturers	386	19	12	6	75	70
Cheap flight tickets from Mumbai to Chennai	399	13	4	1	75	75
List of tropical rainforests in India	912	10	9	8	80	90

Right time to visit shimla	70	4	6	1	70	60
Secrets of Pyramids	72	8	8	5	80	80

$$Precision_r = \frac{|\{\, retrieved\ relevant\ documents\ within\ cut-off\ rank\ r \}|}{Cut\ Off\ Rank\ r}$$

where r is the maximum rank or maximum number of links from the first URL of the search result, accounted for calculating the cut-off precision, in our case we used 20. We observed that the proposed method works well whenever the short queries are used for searching. In our method, many relevant URLs listed in the positions 30 and above in the original query's search result are found projected to the top positions. For example, the query "algorithms to find the prime numbers" does not carry further information like which language, what method, and so on. In this case, it is observed that Google returns many sites showing direct programs instead of algorithms. Our method shows many algorithms compare to programs. We found that the semantic meaning of the given query have further refined to match the maximum related documents. The comparison graph of our method with Google in terms of the effectiveness in search results which we derive is shown in Figure 4.

Though the proposed system performed well in many queries, both in terms of disambiguating the ambiguous words present in the query, it degrades in some occasions. To discuss the possible failure, let us take the example query "Speed of Jaguar". This query would mean Jaguar as animal, or Car, or any entities in the world. For the context involved in this query, the query is correct in the following senses; speed of jaguar as animal and speed of jaguar as car. In this special case, it is difficult to identify one context without further information like what the user mean by Jaguar. Apart from this drawback, our approach is not able to identify that the Jaguar would be a car, as because it is not mentioned in WordNet vocabulary. Hence we would like to include the concept of Named Entity Recognition [22, 23] to identify the named entities, and POS tagging to ensure the order in which the query presents its contextual meaning.

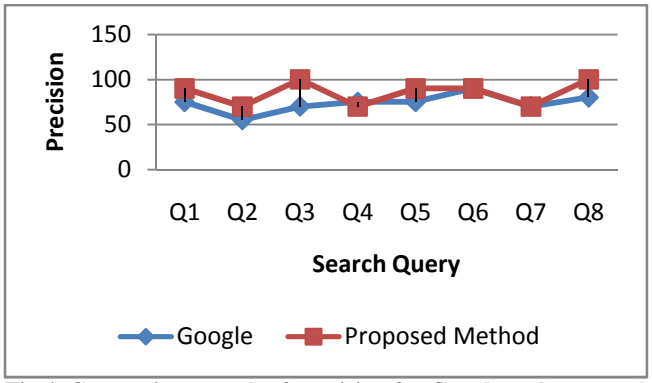

Fig 4. Comparison graph of precision for Google and proposed method

V. Conclusion

In this study, we analyzed the existing models and concluded that most of the search engine users try different possible combination of words to construct a query. Because, there are many number of possible ways to represent a concept through query. We took that as the main point to devise different queries from a single given query. The problem is approached by finding out the various possible queries using thesaurus. Our system compares the search results of various similar queries and calculates new ranks for the result URLs thereby improving the accuracy of the top ranked pages in many queries. One of the important characteristic of our method is that it does not expect any external intervention. Our method could be used in various applications that involve searching information where the user does not have any command over the use of word as per the context. We found that the proposed approach performed effectively in Semantic based Information retrieval, compared to the conventional search systems.

References

[1] Angel F. Zazo, Carlos G. Figuerola, Jose´ L. Alonso Berrocal, & Emilio Rodrı´guez (2005). Reformulation of queries using similarity thesauri. *Information Processing and Management*, 41(5), (1163–1173).

[2] Buckley C., & Voorhees, E. M. (2000). Evaluating evaluation measure stability. *SIGIR 2000*, (pp. 33-40).

[3] Catarina Silvatt, & Bemardete Ribeirot (2003). The importance of stop word removal on recall values in text categorization. *International Joint Conference on Neural Networks,* (pp. 1661 – 1666).

[4] Chakraborty U.K., Chowdhury S., & Roy S. (2010). An adaptive scheme for handling deletion spelling errors' for an intelligent e-learning system. *2nd International Conference on Computer Technology And Development (ICCTD), 2010,* (pp. 260 – 263).

[5] Che-Yu Yang, & Hua-Yi Lin (2010). An automated semantic annotation based-on WordNet ontology. *Sixth International Conference on Networked Computing and Advanced Information Management (NCM)*, (pp. 682 – 687).

[6] Christiane Fellbaum (1998, ed.) WordNet: An Electronic Lexical Database. *Cambridge, MA: MIT Press*.

[7] Gloria Bordogna, Alessandro Campi, Giuseppe Psaila, & Stefania Ronchi (2012). Disambiguated query suggestions and personalized content-similarity and novelty ranking of clustered results to optimize web searches. *Information Processing and Management,* 48(3), (419–437).

[8] Ion Androutsopoulos, & Prodromos Malakasiotis (2010). A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research*, 38(1), (135 – 187).

[9] Jibran Mustafa, Sharifullah Khan, & Khalid Latif (2008). Ontology based semantic information retrieval. *4th International IEEE Conference Intelligent Systems*, (pp. 14-19).

[10] Kongkachandra R, & Chamnongthai K. (2007). Using linguistics information for improving the sentence–based semantic relatedness measurement. *International Symposium on Communications and Information Technologies (ISCIT'07)*, (pp. 1372 – 1376).

[11] Lioma, C. & Ounis, I. (2008). A syntactically-based query reformulation technique for information retrieval. *Information Processing and Management*, 44(1) (143–162).

[12] Lior Meister, Oren Kurland, & Inna Gelfer Kalmanovich (2011). Re-ranking search results using an additional retrieved list. *Journal of Information Retrieval*, 14(4), (413—437).

[13] Ming-Yen Chen, Hui-Chuan Chu , & Yuh-Min Chen (2010). Developing a semantic–enable information retrieval mechanism. *Expert system with Applications*, 37(1) (322-240).

[14] Pablo Castells, Miriam Fernandez, & David Vallet (2007). An adaptation of the vector-space model for ontology-based information retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 19(2), (261-272).

[15] Ricardo Baeza-Yates, & Berthier Ribeiro-Neto (1999) *Modern Information Retrieval*.

[16] Saravanakumar, K. & Deepa, K. (2011). Alternate query construction agent for improving web search result using wordnet. *2011 International Conference on Computational Intelligence and Communication Systems (CICN 2011)*, Oct 2011, (pp. 117-120).

[17] Sathya, M., Jayanthi, J. & Basker, N. (2011). Link based k-means clustering algorithm for information retrieval. *International Conference on Recent Trends in Information Technology (ICRTIT)*, June 2011, (pp. 1111-1115).

[18] Shailesh S. Deshpande, Girish Keshav Palshikar, & G. Athiappan (2010). An unsupervised approach to sentence classification. *International Conference on Management of Data (COMAD 2010)*, (pp. 88 – 99).

[19] Siddharth Patwardhan, & Ted Pedersen (2006). Using wordnet-based context vectors to estimate the semantic relatedness of concepts. *11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*, (pp. 1 – 8).

[20] Ying Liu, Bridget T. McInnes, Ted Pedersen, Genevieve Melton-Meaux, & Serguei Pakhomov (2012). Semantic relatedness study using second order co-occurrence vectors computed from biomedical corpora, UMLs and WordNet. *2nd ACM SIGHIT International Health Informatics Symposium*, (pp. 363 – 372).

[21] Ziqiang Li, & Mingtian Zhou (2010). Use semantic meaning of coreference to improve classification text representation. *2nd IEEE International Conference on Information Management and Engineering (ICIME)*, (pp. 416 – 420).

[22] Hsin-Hsi Chen, Yung-Wei Ding & Shih-Chung Tsai (1998). Named Entity Extraction for Information Retrieval. *International Journal of Computer Processing of Oriental Languages,* 11(4).

[23] M. Khalid, V. Jijkoun, and M. de Rijke (2008). The Impact of Named Entity Normalization on Information Retrieval for Question Answering. *30th European Conference on Information Retrieval (ECIR 2008)*, LNCS 4956, (pp. 705-710).

[24] Freitas, A., Oliveira, J.G., Curry, E., O'Riain, S., Pereira da Silva, J.C. (2011). Treo: Combining Entity-Search, Spreading Activation and Semantic Relatedness for Querying Linked Data. 1st Workshop on Question Answering over Lined Data (QALD-1).

[25] Ch. Aswani Kumar, M. Radvansky, and J. Annapurna. Analysis of Vector Space Model, Latent Semantic Indexing and Formal Concept Analysis for Information Retrieval, *Cybernetics and Information Technologies,* 12(1), 34-48, 2012.