Design and Implementation of a Novel Webpage Ranking Algorithm for improved Web Search

G.S Vinothkumar[1], j.jamet[2], N.Kamal[3]

[1]Assistant Professor in Department of Inforamtion Technology in Sri Ventateswara College Of Engineering & Technology

[2]Professor and Head in the Department of Computer Science & Engineering in Sri Venkateswara College Of Engineering & Technology

[3]Professor and Head in the Department of Information Technology, in Sri Venkateswara College of Engineering & Technology

Abstract: With the rapid growth of the Semantic web and its critical role in providing the needy information, proves its significance as the information provider. Thus the most critical task for the search engines is to report with the relevant information at the right time. Semantic web uses ontology's to represent the knowledge information. Default general purpose approaches are generally adopted to rank the resultant pages, but do suffers with certain irrelevant results. This irrelevancy may occur due to the improper or poor ranking measures. Several search engines have been proposed in the due course but most of them failed to consider the key aspect of the semantic web which is called as Relation. In this paper, we first discuss about the ranking methods those have been proposed earlier, exploring their features, analyze their strengths and weaknesses and also propose a new set of ranking measures with different set of sophisticated features. We use certain measures such as Depth measure, Concept match measure, Semantic similarity measure, Inter-relation measure etc. to evaluate the relevancy of the page. Our ranking measure produces more effective results and thus meets the needs of the users.

Keywords: Depth measure, Semantic similarity measure,motivation,preprocessing

Reference to this paper should be made as follows: G.S Vinothkumar[1], j.jamet[2], N.Kamal[3] (2014) 'Design and Implementation of a Novel Webpage Ranking Algorithm for improved Web Search', *International Journal of Inventions in Computer Science and Engineering*, Volume 1 Issue 3 2014.

1 Introduction

Semantic Web is the extension of World Wide Web, which helps us in improving the clarity of the information. It further provides a mechanism to better understand the needs of the users in by way of effective knowledge representation. It contributes several mechanisms which are used to classify information based on its context results in retrieving relevant information on web. It contains numerous Resource Framework Descriptions (RFD`s) which are used for effective knowledge representation.

Ontology is a formal, explicit specification of shared conceptualization [7]. There are various ontology libraries are search engines which are widely used. These facilitate in representing applications which are in vogue through effective Domain related ontologies. Standard web ontology language (OWL) based on RDF model is used to describe the concepts with their relationship

The main drawback in the ontology construction is the high cost involved in ontology construction. More time is required to gather complete information pertaining to a specific domain. Thus, the performance outcome of the resulting ontology is not known. There exists no mechanism of fixed theory to ensure that the resulting ontology will be better than the existing one. Therefore, we hereby propose an approach to construct new ontologies in which we can reuse the existing ontologies.

The search engines aids in effective retrieval of the information, but it is known fact that the retrieved web pages may also contain ineffective or irrelevant information which are unavoidable. Semantic web provides a means in overcoming this problem by implementing the rank algorithms. The ranking algorithm works effectively by identifying the quires of the user through search engines and provides the desired result, we use most of the existing solutions which are needed to work on the whole annotated knowledge to rank the results.

The search engines uses domain ontology to retrieve the web pages which contains domain related information. These search engines follow some regular methods. This produces irrelevant results and the web page which contains irrelevant information is rated top rank and occurs in the very first page or in the subsequent pages. This restricts the users from visiting those web pages which are closely relevant. Search engines like Google uses click streams as a ranking measure which ensures that the page which is visited frequent number of times is ranked as top. Some times a page which doesn‟t have any relevant information

may be visited by the user, due to the appearance of the first page, but still will be top ranked due to frequent user clicks. This type of ranking does not measure the actual popularity of the web page and thus considered as a worst type of ranking. We thus need to propose certain unique measures to perform honest ranking of the web pages. We hereby discuss some of the new ranking measures in the following sections of this paper.

II. Motivation

Real scenario of World Wide Web contains huge volumes of heterogeneous information organized in structured form. This information tends to change from time to time. In such an environment where information tends to change frequently, searching relevant information is definitely a difficult task for the user. Numerous search engines evolved and are using semantic ontology to accelerate the user search. Selection of an optimal ranking strategy is essential while implementing semantic web search engines. While the existing ranking measures may not provide promising results, the occurrence of more pressing situations motivated us to find new ranking measures.

World Wide Web is a dynamic environment where the data and user changes frequently. It contains huge volumes of structured and unstructured data. In such a dynamic environment, quick access of desire information becomes crucial. Tracking and identifying user search behavior is also difficult. This motivated towards personalization of web search where the users‟ search environment can be customized based on their interests. This approach thus aids the user to identify their information need without much difficulty. Semantics plays an effective role in web search to provide exact and accurate results in a more effective and efficient way. Semantics can be used to identify and categorize the web pages based on the topic and further categorizes the search query and the page visited.

III.Related Work

Context-Aware Semantic Association Ranking [21] enhances the performance od Semantic Web search engines using enhanced ranking measures. The paper proposes a similarity score which is the difference between the query described and the retrieved resources [21]. Initially the query is analyzed and the hidden relation from the query is extracted. This explains the initial set of relation which can inferred from the query. The similarity value is computed which depicts the ratio between the user query and the relation instances. This computation is performed to all the properties of the web semantic instances. Similar approach, aimed at measuring the relevance of a semantic association(that is, a path traversing several concepts linked by semantic relations) is illustrated in Ontology-based lexical relations like synonyms[18], antonyms and homonyms between keywords (but not concepts) have been used to “expand” query results. In this case search is targeted to the web, rather than to the semantic web.

In [27], a similar approach has been integrated into artificial intelligence methodologies to address the problem of query answering. In query logs are used to construct a user profile to be used later to improve the accuracy of Web search. Semantic Web search from the user‟s point of view has also been addressed in [15] and [28], where the authors present two methodologies for capturing the user‟s information needs by trying to formalize its mental model. They analyze keywords provided during query definition, automatically associate the related concepts, and exploit the semantic knowledgebase to formulate formal queries automatically.

Automatic evolution of search engines via implicit feedback is proposed in [1]. Here, the ranking of pages is done based on the implicit feedback from the users. In this method, they collect the user feedback from various user actions like save, copy, print, bookmark, etc. The user actions are tracked in the background and stored. During the ranking process these user feed backs are used to calculate the weight for a particular web page. According to the calculated weight, the web page is ranked and returned as result.

A slightly different methodology has been exploited in SemRank [2]. Here, the query results are ranked based on the reliability, genuineness of the web pages which ensures that the user is catered with all the relevant information that he actual needs. To achieve their goal, the authors define two measures, namely “uniqueness” and “discrepancy” which accounts specificity or deviation of a particular result with respect to the instances stored in the database. In the rank computation, we exploit “Modularize Relevance Model”, that accounts the particular context/purpose in/for which a query has been submitted (conventional or discovery search). An additional value of SemRank is introduced in Modularize Relevance Model. The authors did not provide any methodology to compute the cost of their approach. It is a fact that, we may need information related to the annotations of all the remaining pages, to rank even a single page. Hence, the performance of the proposed solution would hardly scale for huge Semantic Web environments.

An TF and IDF page ranking measure is described in XSearch: A semantic search engine for xml[12]. In XSearch, sub trees of a document are returned to the user. Hence, they compute the weights of the keywords at a lower granularity, i.e., at the level of the leaf nodes of a document. This allows the algorithm to determine those sub trees of a document which are more relevant.

Swoogle: A Search and Metadata Engine for the Semantic Web [3], describes a ranking algorithm. In this, the system crawls the web pages and indexes them into the system for the semantic web. It analyzes the web page and its metadata and extracts the keywords. Metadata of the web page or documents are used to compute the relation between documents. The related documents are indexed into the system for future retrieval. The Ngrams from the document relevant to the keyword is used to compute the similarity

between the documents and query string. In this paper, ranking is done based on the relative importance of the web documents. The relative importance is calculated using the probability of the direct user arrival to the page or the probability to arrive to the link through some other page. Unfortunately, this random surfing model is not appropriate for the Semantic Web. The semantics of links lead to a non-uniformed probability to follow a particular outgoing link.

In a Relation-Based Page Rank Algorithm for Semantic Web Search Engines [10], the author proposed a ranking algorithm, which is based on the similarity value calculation done by page sub graph and query graph. Here, the key words from all the documents are extracted. A page graph is constructed for each web document. Query sub graph is generated using the query terms. Page forest is computed and using this, page score for each page is calculated by connected graph.

III. Proposed Methodology

In the proposed method, we identify the concept class of the query. Further, we compute all the ranking parameters for each page and return the results based on the scores. We submit the query to the semantic search engine and retrieve the result from it. Upon receiving the query result, we store the query results and download all the resultant url‟s, store them into local storage. Then our process starts in different stages as follows.

Concept Measure

In Concept Measure, the algorithm searches the class labels in the domain ontology for matching the query wholly or partially. The ontology classes would be ranked higher if it matches all the query terms. The terms used are Ec denotes the total number of classes that matches exactly the term „T‟. „o‟ be the particular ontology belongs to the set „O‟. Pc denotes the total number classes that matches partially with the term „T‟.

CM[o,T]=Ec[o,T]*μ+Pc [o, T] * v, 0< μ, v<1 (3)

Preprocessing

At this stage we read all the text information from the downloaded document. We remove all stop words from the page content and we perform stemming process. Finally we get set of keywords from the web document. This process is carried out for each web document downloaded as query result. The same stop word removal and stemming processes are carried out in the query text also. Then we identify the domain or concept of the query and we retrieve the keywords from the concept or domain ontology.

Algorithm1:

Start Receive user query
Submit query to search engine
Receive results from search engine.
For each result link
Download all web documents.
Extract text from web document.
For each term from collection
Check for stop word Perform stemming process
End
End
Identify query domain
Retrieve domain ontology from database.
Extract concepts and classes from ontology.
Extract relations from ontology.
End

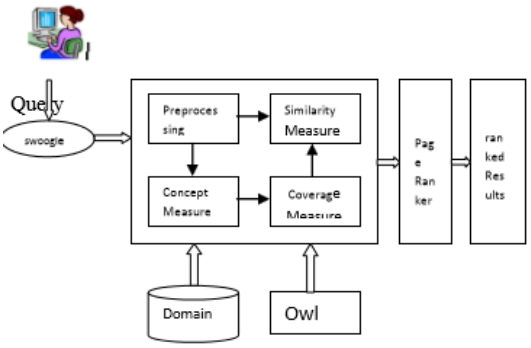

Fig1: shows the architecture of the system

Concept Measure

In Concept Measure, the algorithm searches the class labels in the domain ontology to match the user query either exactly or partially. If the ontology classes match all the query terms then it will be ranked higher. The term Ec denotes the total number of classes that matches exactly with the term „T‟. „o‟ be the particular ontology belongs to the set „O‟. Pc denotes the total number classes that matches partially with the term „T‟.

CM[o,T]=Ec[o,T]*μ+Pc [o, T] * v, 0< μ, v<1 (3)

Coverage Measure The density measure is calculated to identify the coverage factor of the document for the particular concept. The more it contains the related terms, the more it covers the concept. This is calculated as follows. Tc- Number of terms in the Concept. Tn- Number of terms in the document.

Cov=Tn/Tc (4)

Similarity Measure

The similarity between the pages is calculated using term frequency and inverse document frequency. The term frequency is calculated using the number of times it occurs in the corpus and total number of terms occurs in the corpus. Inverse document frequency is calculated using number of documents it occurs in other documents and total number of documents. Then we calculate the weight factor by multiplying the tf and idf factors. This weight value represents the relationship of a particular page with the other pages.

Algorithm 2:

For each term in document
Calculate term frequency using 1.
Tf = tn/total; ----(1).
tn-total number of times the term occurs.
Total- total number of terms in the document.
Calculate Inverse Document frequency
Idf=To/Total ----(2).
To-total number of times the term occurs in other documents.
Total-total number of documents.
Calculate weight for each term using
W=tf*idf;
End
Similarity Measure weight Sm=Sum all weights ∑w ---(3);

Feedback Measure: We collect the implicit feedback from the user by tracking the user behavior on the web page. We track user actions like save, copy, bookmark and store in the data base. Based on the results, we can have tracked result for a particular web page. We calculate the measure or we assign it to zero. The calculated measure will be used to rank the web page.

Algorithm 3:

For each url from the result
Retrieve actions performed from the database
Calculate Ta-Total sum of number of actions.
Calculate Tv-Total sum of number of visits.
Calculate the feedback weight of urls.
Fw=Ta/Tv
End
Page Ranking:
The page ranking is performed according to the rank score of the url.
We calculate the rank score by using the following algorithm.

Algorithm 4:

Start
For each url
Rank Score rs=CM+DW+Cov+Fw;
End
Sort the url according to Rank Score rs;
Pop the ranked result.
End

In this algorithm we calculate the rank score of each url , using Concept measure, Coverage measure, Similarity measure and Feedback measure. The rank score is sorted and the url of more rank score tops the list. All the urls are sorted in descending order of the page rank score. We form the list of urls by sorting the page rank score. The sorted url list is returned as ranked result to the user.

V.Results And Discussion

Here we discuss about the results produced by our methodology.

Fig:2 shows the result for data mining.

Fig:3 shows the web page viewed

Fig 4. Shows the result of user action

Fig 5. Shows the ranked results

Conclusion

The paper describes about the web page ranking algorithm used to rank the web pages in semantic web to give the best relevant results to the user. The experiment is done by using Google search engine and domain ontology and the results are highlighted. Thus the result shows an improved score compared to an existing algorithm. This helps the researchers to sort out the best ranking measure for reusing purpose.

References

[1] B. Aleman-Meza, C. Halaschek, I. Arpinar, and A. Sheth, "A Context-Aware Semantic Association Ranking," Proc. First Int‟l Workshop Semantic Web and Databases (SWDB ‟03), pp. 33- 50, 2003.
[2] K. Anyanwu, A. Maduko, and A. Sheth, "SemRank: Ranking Complex Relation Search Results on the Semantic Web," Proc. 14th Int‟l Conf. World Wide Web (WWW ‟05), pp. 117-127, 2005.
[3] R. Baeza-Yates, L. Caldero´n-Benavides, and C. Gonza´lez-Caro, "The Intention behind Web Queries," Proc. 13th Int‟l Conf. String Processing and Information Retrieval (SPIRE ‟06), pp. 98-109, 2006.
[4] T. Berners-Lee and M. Fischetti, Weaving the Web. Harper Audio, 1999.
[5] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," Scientific Am., 2001.
[6] S. Brin and L. Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine," Proc. Seventh Int‟l Conf. World Wide Web (WWW ‟98), pp. 107-117, 1998.
[7] S. Cohen, J. Mamou, Y. Kanza, and Y. Sagiv, "XSEarch: A Semantic Search Engine for XML," Proc. 29th Int‟l Conf. Very Large Data Bases, pp. 45-56, 2003.
[8] L. Ding, T. Finin, A. Joshi, R. Pan, R.S. Cost, Y. Peng, P. Reddivari, V. Doshi, and J. Sachs, "Swoogle: A Search and Metadata Engine for the Semantic Web," Proc. 13th ACM Int‟l Conf. Information and Knowledge Management (CIKM ‟04), pp. 652-659, 2004.
[9] L. Ding, T. Finin, A. Joshi, Y. Peng, R. Pan, and P. Reddivari, "Search on the Semantic Web," Computer, vol. 38, no. 10, pp. 62-69, Oct. 2005.
[10] L. Ding, P. Kolari, Z. Ding, and S. Avancha, "Using Ontologies in the Semantic Web: A Survey," Ontologies, pp. 79-113, Springer, 2007.
[11] R. Guha, R. McCool, and E. Miller, "Semantic Search," Proc. 12th Int‟l Conf. World Wide Web (WWW ‟03), pp. 700-709, 2003.
[12] Z. Gyongyi and H. Garcia-Molina, "Spam: It‟s Not Just for Inboxes Anymore," Computer, vol. 38, no. 10, pp. 28-34, Oct. 2005.
[13] C. Junghoo, H. Garcia-Molina, and L. Page, "Efficient Crawling through URL Ordering," Computer Networks and ISDN Systems, vol. 30, no. 1, pp. 161-172, 1998.
[14] S. Kapoor and H. Ramesh, "Algorithms for Enumerating All Spanning Trees of Undirected and Weighted Graphs," SIAM J. Computing, vol. 24, pp. 247-265, 1995.
[15] Y. Lei, V. Uren, and E. Motta, "SemSearch: A Search Engine for the Semantic Web," Proc. 15th Int‟l Conf. Managing Knowledge in a World of Networks (EKAW ‟06), pp. 238-245, 2006.
[16] Y. Li, Y. Wang, and X. Huang, "A Relation-Based Search Engine in Semantic Web," IEEE Trans. Knowledge and Data Eng., vol. 19, no. 2, pp. 273-282, Feb. 2007.
[17] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," Stanford Digital Library Technologies Project, 1998.
[18] A. Pisharody and H.E. Michel, "Search Engine Technique Using Keyword Relations," Proc. Int‟l Conf. Artificial Intelligence (ICAI ‟05), pp. 300-306, 2005.
[19] T. Priebe, C. Schlager, and G. Pernul, "A Search Engine for RDF Metadata," Proc. 15th Int‟l Workshop Database and Expert Systems Applications (DEXA ‟04), pp. 168-172, 2004.
[20]H.KnublauchProte´ge´,StanfordMedicalInformatics,http ://protege.cim3.net/file/pub/ontologies/tra vel,2002.
[21] Automated Evaluation of Search Engine Performance via Implicit User Feedback
[22] XSEarch: A Semantic Search Engine for XML
[23] Swoogle: A Search and Metadata Engine for the Semantic Web

G.S.VinothKumar receive his M.Sc. Degree in Computer Science from Madras University and M.E Degree in Computer Science & Engineering from Sathyabama University. He is working as Assistant Professor in Department of Information Technology in Sri Venkateswara College Of Engineering & Technology, Chennai. His Research interest includes Data Mining and Mobile Adhoc Networks.

Dr.Janet.J received her B.E. and M.E Degree in Computer Science & Engineering from Madras University. She has completed his PhD from Sathyabama University, Chennai. Now she is working as a Professor and Head in the Department of Computer Science & Engineering in Sri Venkateswara College Of Engineering & Technology, Chitoor, Andhra Pradesh. Her research interests include Image Processing, Data Mining, and Networks.

Dr.Kamal.N received his B.E Degree in Computer Science & Engineering from Madras University and M.E Degree in Computer Science & Engineering from Sathyabama University. He has completed his PhD from St.Peter‟s University. He is working as a Professor and Head in the Department of Information Technology, in Sri Venkateswara College of Engineering & Technology, Thiruvallur Chennai. His research interests include Mobile Ad-hoc networks, Vehicular Ad-hoc networks, Sensor Network.