ISSN No. 0976-5697

Volume 5, No. 3, March-April 2014

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Ranking Relevant Web Pages using Ontology

Dr. S.S. Dhenakaran
Asst. Prof. of Computer Science
Alagappa University
Karaikudi, Tamilnadu, India

S.Yasodha*
Asso. Prof. of Computer Science
Govt. Arts College(W),
Pudukkottai, Tamilnadu, India

Abstract: Conventional search engines like Google and Yahoo rely on keywords for searching and they fail to consider the semantics of the query. This leads to irrelevant ranking of Web pages. More sophisticated methods that do provide the relevant information for the query need to be designed. The Semantic Web that stores metadata as ontology could be used for this purpose. This paper proposes an ontology based framework for ranking Web pages. The proposed framework combines the Vector Space Model of Information Retrieval with Ontology. The framework constructs semantically annotated RDF (Resource Description Framework) files which form the RDF knowledgebase for each query which is used to rank the Web pages. The proposed framework has been evaluated by two measures, precision and relative recall.

Keywords: Ontology, Resource Description Framework, Knowledgebase, Semantic Web

I. INTRODUCTION

The Web is more than just a collection of textual documents. For information retrieval from the Web, users currently rely on human-generated semantic markup. Even though such markup languages provide a uniform framework for interchanging data and metadata between applications, they do not provide any means of considering the *semantics* of data.

Moreover, the tremendous growth of Web has made information retrieval a time-consuming, tedious and a boring process. Not only the relevant information but also the irrelevant information for a user's query is returned which turned browsing a time-consuming process. Moreover, the generic search engines like Google are keyword-based. They fail to consider the semantics of the keywords. So, handling keywords with multiple semantics is often an omitted task of search engines. For example, the keyword *Principal*, would mean *The head of the institution* in one context and *The amount invested* in another context. This disparity could not be dealt-with by search engines and they provide information related to both contexts when the term *Principal* is given as search keyword.

Another problem with search engines is the lack of very strong anti-spamming mechanisms. The keyword-based indexing of search engines paved way for malicious Web spamming. So, the relevant websites are not ranked in top-order. It is a human tendency to trust and follow only a few top-ranked websites, which degrades the quality of search engines. The aforesaid problems could be overcome to some extent by a few statistical algorithms and relevance feedbacks that filter the search results. But they too fail to provide the most relevant search results.

So, the Semantic Web has emerged which tries to solve these problems and do provide the most relevant results for the users' query. In the Semantic Web, the semantic metadata of each page is stored along with the contents of the Web page. The semantics of the different terms in a particular domain are provided as ontology. So ontology-based frameworks need to be designed that possess knowledge about the user query, annotated Web pages and the underlying ontology.

Four different types of technologies are available for building the Semantic Web: Metadata, Ontology, Logic and Agents. In this paper an ontology-based framework for ranking Web pages has been proposed and implemented and tested. This framework is implemented in JAVA and ontology engineering is done using RDF (Resource Description Framework). The screen shots are designed using Net Beans IDE. The performance of the framework is evaluated using two metrics, precision and relative recall.

II. PREVIOUS RESEARCH

The rapid growth of Web and the increasing demand has made information retrieval, a difficult task. The users are looking for more efficient information retrieval mechanisms and tools for finding, filtering and extracting the necessary information. Hundreds of search engines are available, but only a few like Google and Yahoo are popular because of their crawling and ranking methodologies. So designing efficient Web mining and ranking mechanisms is very necessary for effective information retrieval [1].

The pitfalls in today's search engines could be eliminated by implementing context-aware semantic search engines. The authors proposed, designed and implemented a semantic search engine named SIEU (Semantic Information Extraction in University Domain) and tested with the University domain [2]. A relation-based page rank algorithm was proposed by the authors and they used it in conjunction with semantic web search engines that extract information from user queries and annotated resources. The performance analysis was done by measuring relevance score [3].

A link-editing algorithm based on relative page popularity was proposed that could automatically revise a website's page structure to develop an effective information retrieval system. The common notion of web access is that "Faster is the access, better is the organization of the web server". But in this paper, the authors say that for commercial websites, the best organization of web server may be the one that achieves the highest AA (Absolute Access per page) [4]. Semantic Information Retrieval has become the crucial component of search engines. Ontology-based Semantic Web Search (SWS) research is at its peak.

© 2010-14, IJARCS All Rights Reserved

DySE (Dynamic Semantic Engine) [5] implements a context-driven methodology, in which keywords are split into subject keywords and domain-specific keywords. A dynamic system is used that constructs ontology dynamically and uses that as a knowledge base. The procedure for representing natural language queries as semantic networks is proposed in [6]. A syntactic analysis of the query is done by parsing the query using Stanford Parser to tag words with the corresponding parts of speech.

SocialPageRank algorithm [7] is based on the observation that the popularity of users, resources and tags within a folksonomy are highly interdependent. For example, the popularity of resources is high when they are annotated by more number of users with popular tags. On the other hand, the popularity of tags is high when more number of users attach them to popular resources. The PageRank algorithms proposed in [8] namely SocialSimRank and FolkRank are based on random surfer model. But the difference between the algorithms relies on the types of links that are followed by the "random surfer". SocialPageRank restricts the "random surfer" to paths like resource-user-tag, whereas FolkRank is more flexible and allows paths like resource-tag-resource.

SemRank[9] ranks results based on its predictability to the user. The information content is measured by its specificity and the deviation of a particular result. The drawback of the algorithm is that for ranking a single page, the remaining pages have also to be considered. Context-based keyword search is proposed in [10]. User defined weights are assigned to each semantic association. The measures being considered are trust value, path length and specificity.

Though earlier researchers have developed a plethora of algorithms and methodologies, it is imperative to find new algorithms. Moreover, all these methods and algorithms fulfill the objectives of semantic web mining to some extent. But still they have some flaws which are to be rectified. So efficient methods have to be designed.

III. ONTOLOGY

The term ontology denotes a formal and explicit specification of a shared conceptualization. Ontology includes terms and their relationships. The term denotes important concepts of the domain. For example, in a university domain, students, courses, faculty members, and disciplines are some of the concepts. The *relationships* denote *hierarchies of classes.* Ontologies are helpful for the navigation and organization of Websites. They are also helpful for increasing the precision of Web searches.

There are four important components of ontology. They are:

- ***a.*** ***Concepts–*** A *concept* denotes a set or class of entities or `things' within a domain. For example, Vice-Chancellor is a concept within the domain of University.
- ***b.*** ***Relations–*** *Relations* indicate the interactions between concepts or a concept's properties. For example, *Vice- Chancellors areappointedby the Governor.*
- ***c.*** ***Instances–*** *Instances* are the `things' indicated by a concept. For example, *Malala is an instance of the concept student.*
- ***d.*** ***Axioms–*** *Axioms* are used to constrain values for classes or instances. For example, *Students securing less than 50% of marks should reappear.*

IV. RESOURCE DESCRIPTION FRAMEWORK (RDF)

RDF is a World Wide Web Consortium (W3C) specifications originally designed as a metadata data model. RDF is a foundation for processing metadata; it provides interoperability between applications that exchange machine-understandable information on the Web. It stores metadata about files and other machine-accessible resources. RDF documents consist of three types of entities:

- ***a.*** ***Resources*** **-** Resources may be Web pages, parts or collections of Web pages, or any real-world objects that are not directly part of the WWW. In RDF, resources are always addressed by URIs.
- ***b.*** ***Properties*** **-** Properties are specific attributes, characteristics, or relations describing resources.
- ***c.*** ***Statements*** **–** Each statement consists of (*Resource, Property, Value*) triples. In the RDF graph example shown in Figure 1,

  Dhoni is a **resource**
  <plays> is a **property**
  The string *« Cricket »* is a **value.**

RDF can be used in a variety of application areas; for example: in resource discovery to provide better search engine capabilities, in cataloging for describing the content and content relationships available at a particular Web site, page, or digital library, by intelligent software agents to facilitate knowledge sharing and exchange, in content rating, in describing collections of pages that represent a single logical "document", for describing intellectual property rights of Web pages, and for expressing the privacy preferences of a user as well as the privacy policies of a Web site. RDF with digital signatures is the key in building the "Web of Trust" for electronic commerce, collaboration and other applications.

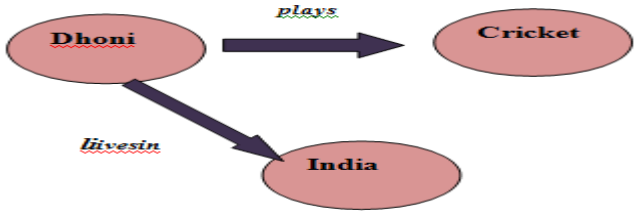

Figure 1: RDF Graph Example

V. METHODOLOGY

We propose a new framework named ONTOPARK for ranking relevant Web pages. ONTOPARK is an Ontology-Based Page Ranking framework using RDF. The proposed framework is an extension of the traditional Vector Space Model of information retrieval. It is combined with ontology, the Semantic Web technology that enables meaningful information retrieval from the Web. The framework design in shown in figure 2. The framework works in three phases: Preprocessing, Ontology Construction and Ranking.

A. Phase I - Preprocessing:

In this phase, the framework accepts the query from the user and extracts Web links from Web database. Then it

© 2010-14, IJARCS All Rights Reserved

preprocesses the query as well as the snippets and contents of each Web page by applying preprocessing steps like Stopwords removal, Stemming and Parts-of-Speech tagging.

a. ***Stopwords removal-*** Stopwords are insignificant words that appear frequently in queries. Articles, Prepositions, pronouns and conjunctions are the commonly occurring stopwords. For example words like "*a, about, an, are, by, from, how, on, of, that, these, the, this, was, when, who, where, with* etc" are stopwords. Such insignificant words are removed from the query.

b. ***Stemming -*** Stemming refers to the process of suffix removal. In the proposed framework, stemming is done by Porter stemmer method. The Porter stemmer is a process for eliminating the inflected endings from words in English. For example, the words *talk, talking* and *talkative* are reduced to their root word *talk* by stemming.

c. ***POS-tagging:*** POS-tagging is the process of tagging up a word in a text to a particular part of speech such as noun, verb, adverb, adjective etc. For example, the phrase "Cook meat in a big vessel" is tagged as:

Word	**Tag**
cook	verb (noun)
meat	noun
in	preposition (noun, adverb)
a	determiner (noun)
big	adjective (noun)
vessel	noun

The proposed framework uses Stanford POS tagger for POS Tagging. The refined query obtained in this module is used as input for the next module.

B. Phase II –Ontology Construction:

After preprocessing the query, snippets and the contents, RDF knowledge base is constructed for each query. RDF files are created for the Web pages whose page rank of Google in non-zero. The RDF files are created by combining the Web link, title, preprocessed snippet and the preprocessed contents corresponding to each Web link. The collection of these RDF files forms the RDF knowledgebase for that query. This RDF knowledge base is used in the next phase for ranking.

C. Phase III - Ranking:

Ranking is based on the adaptation of the Vector Space Model of information retrieval. In the Vector Space Model, term weights are computed for query terms by counting the number of occurrences of the term in the documents of the Web database. But in the proposed framework, term weights are computed for query terms that appear in the RDF files of the RDF knowledgebase. Term weight is computed by an adaptation of the TF-IDF algorithm, where TF denotes the Term frequency and IDF denotes the inverse document frequency. Using this term weight, relevance score is computed to measure the similarity of the query to each RDF file in the RDF knowledgebase. Ranking is done based on this relevance score

Consider Knowledge base K with RDF files $r_1, r_2,\ldots r_m$. The framework accepts a query $Q = \{x_1\ldots x_n\}$ containing the terms $\{x_1\ldots x_n\}$. The answer to the query is a list of the top n documents. The term frequency $tf\ (x,r)$ is the number of times that the term x appears in RDF file r. The document frequency $df\ (x,K)$ is the number of RDF files in K that contain x.

The weight $W\ (x,r)$ of a term x in an RDF file r is computed as:

$$W(x,r) = tf(x,r) \ \mathbf{X}\ idf(x,r)$$

Where $tf(x,r)$ is the normalized frequency of term x in RDF file r which is computed as

$$tf(x,r) = \frac{freq(x,r)}{max\{freq(y,r)\}}$$

Where $freq(x,r)$ is the number of occurrences of the term x in r, $max\{freq(x,r)\}$ is the frequency of the most repeated term in RDF file r.

The inverse document frequency $idf(x,r)$ is computed as :

$$idf(x,r) = \log \frac{N}{df(x,r)}$$

Where N is the set of all RDF files in the knowledge base and $df(x,r)$ is the number of RDF files annotated with x. The documents are ranked according to a relevance score Score(Q, r),which is the relevance of an RDF file r to the query Q.

$$Score(Q,r) = \sum_{x \in Q,r} W(x,r).\ln \frac{|K|+1}{df(x,K)}$$

Where $|K| = m$ is the size of the Knowledgebase K.

VI. RESULTS AND DISCUSSION

The efficiency of the framework has been evaluated by two measures: Precision and Relative Recall. Precision is the measure of accuracy. It measures the relevance of Web pages with respect to the total retrieved. Relative Recall measures the quantity of Web pages retrieved with respect to the total available. Average Precision or Average Relative Recall(AP / AR) values are computed as the average of all the precision values or relative recall values respectively. Mean Average Precision or Relative Recall (MAP / MAR) values are computed as the mean of Average Precision and Average Relative Recall values of single word and multi word queries.

$$\text{Precision} = \frac{\text{Total relevant for each query}}{\text{Total retrieved for that query}}$$

$$\text{Relative Recall} = \frac{\text{Total retrieved by ONTOPARK}}{\text{Total retrieved by Google and ONTOPARK}}$$

AP / AR = Average Precision / Relative Recall of Single-Word and Multi-Word queries.

MAP / MAR = Mean of Average Precision/Relative Recall of Single-Word and Multi-Word queries.

The proposed framework is an attempt to combine ontology with the traditional Vector Space Model to retrieve meaningful Web pages. The framework is implemented in JAVA and the screenshots are designed using Net Beans IDE. The framework is tested by a few single word and multi word queries. The performance is evaluated by two metrics precision and relative recall. The results are compared to that of Google. The results have been tabulated in Table 1. The screen interfaces are shown in Fig. 3 and 4.

© 2010-14, IJARCS All Rights Reserved

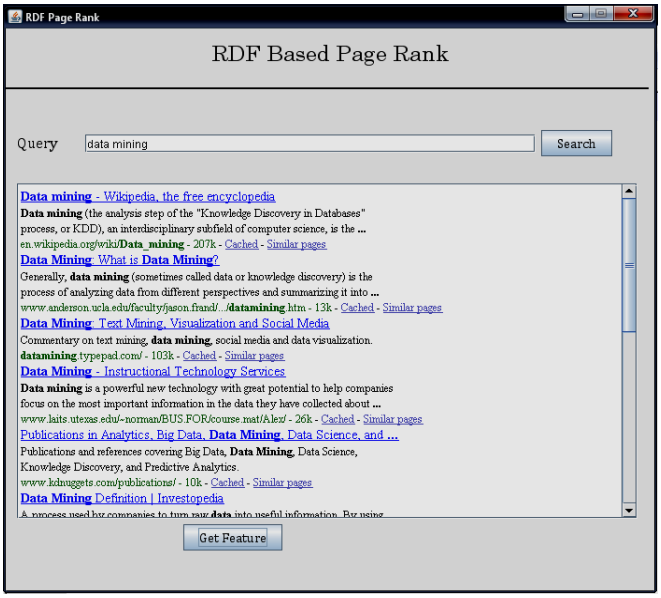

Figure: 3: Sample Screen Interface

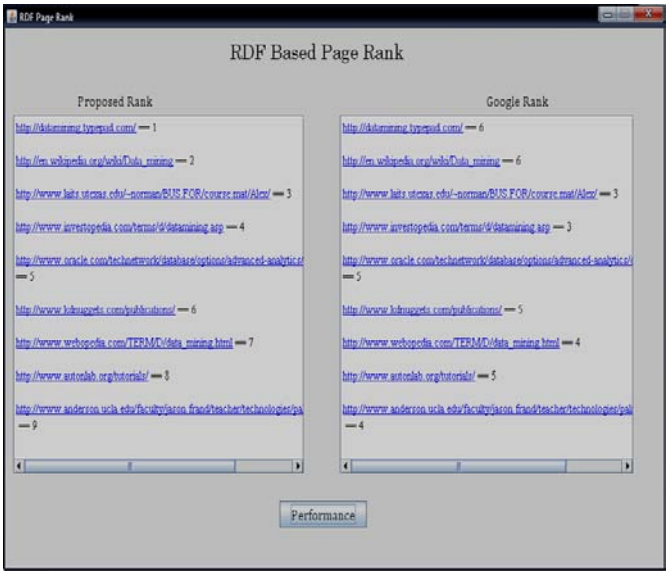

Figure 4 : Sample Screen Interface

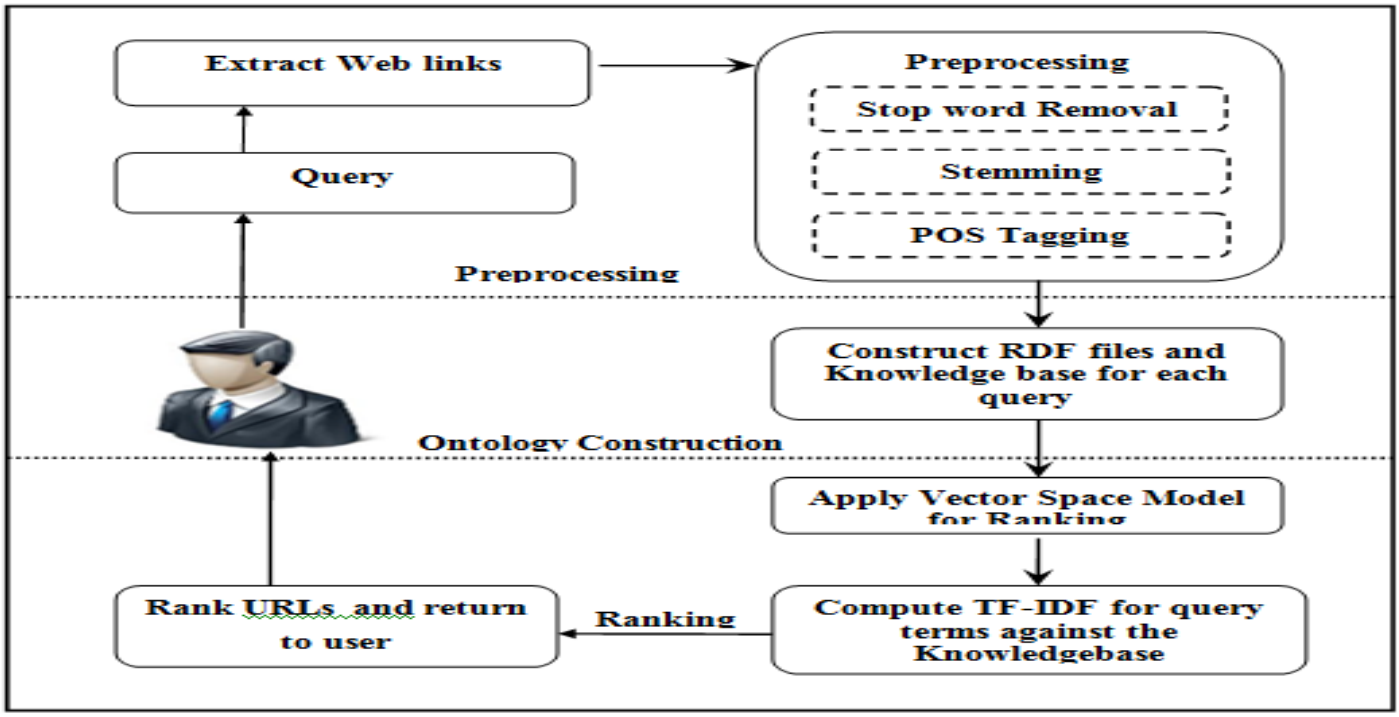

Figure 2 : ONTOPARK Framework Design

Table 1: Mean Average Precision And Recall

	Proposed	Google
Mean Precision of Single Word Query	0.76	0.72
Mean Average Precision	0.80	0.70
Mean Precision of Multi Word Query	0.78	0.71
Mean Relative Recall of Single Word Query	0.47	0.53
Mean Relative Recall of Multi Word Query	0.49	0.51
Mean Average Relative Recall	0.48	0.52

© 2010-14, IJARCS All Rights Reserved

VII. CONCLUSION

The proposed framework was designed as an extension to the traditional Vector Space Model. It was combined with ontology to produce semantic search results. Though the precision of search was increased, there are still limitations with this framework. RDF files are created only for the top 30 Web pages and this has to be increased. There is a long way to go in the area of Semantic Web Mining and research in this particular area should also be encouraged.

VIII. ACKNOWLEDGEMENTS

The author **S.Yasodha** would like to acknowledge University Grants Commission (UGC) of India, for the financial support extended to this project under the Minor Research Project Scheme (LINK F. 3923/11 UGC-SERO).

IX. REFERENCES

[1]. M. G. da Gomes Jr. and Z.Gong, Web Structure Mining: An Introduction, Proceedings of the IEEE International Conference on Information Acquisition, 2005.

[2]. Swathi Rajasurya, Tamizhamudhu Muralidharan, Sandhiya Devi and Dr.S.Swamynathan, Semantic Information Retrieval Using Ontology in University Domain", http://arxiv.org/ftp/arxiv/papers/1207/1207.5745.pdf

[3]. Fabrizio Lamberti, Andrea Sanna, and Claudio Demartini, A Relation-Based PageRank Algorithm for Semantic Web Search Engines, IEEE Transactions on Knowledge and Data Engineering, 2009, **21(1)**: 123-136.

[4]. John Garofalakis, Panagiotis Kappos, and Dimitris Mourloukos, Web Site Optimization Using Page Popularity, IEEE Internet Computing, University of Patras, Greece, 1999.

[5]. Antonio M. Rinaldi, "An Ontology-Driven Approach for Semantic Information Retrieval on the Web", ACM Transactions on Internet Technologies, 2009, **9(3)**:10:1-10:24.

[6]. Joel Booth, Barbara Di Eugenio, Isabel F. Cruz, Ouri Wolfson, Query Sentences as Semantic (Sub) Networks, IEEE International Conference on Semantic Computing, Chicago, USA, 89-92.

[7]. S. Bao, G. Xue, X. Wu, Y. Yu, B. Fei, and Z. Su, Optimizing Web Search using Social Annotations, Proceedings of 16th International World Wide Web Conference (WWW '07), ACM Press, 2007, 501- 510.

[8]. L. Page, S. Brin, R. Motwani, and T. Winograd, ThePageRank Citation Ranking: Bringing Order to the Web, Technical report, Stanford Digital Library Technologies project, 1998.

[9]. K. Anyanwu, A. Maduko, and A. Sheth, Sem Rank: Ranking Complex Relation Search Results on the Semantic Web, Proceedings of 14th International Conference on World Wide Web (WWW '05), 2005,117-127.

[10]. N. Stojanovic, R. Studer, and L. Stojanovic, An Approach for the Ranking of Query Results in theSemantic Web, Proceedings of the Second International Semantic Web Conference (ISWC '03), 2003, 500-516.

© 2010-14, IJARCS All Rights Reserved