MULTI MODEL DYNAMIC WEB CRAWLER WITH HIERARCHICAL SUBSPACE CLUSTERING FOR EFFICIENT WEB SEARCH USING T^2S PAGE RANKING

Kowsalya N
Assistant Professor,
Department of Computer Science,
Vivekanandha College of Arts and Sciences for Women,
Elayampalayam - 636 005, Namakkal
kowsalyaphd123@gmail.com

Dr. C. Chandrasekar, M.C.A., Ph.D.,
Associate Professor, Department of Computer Science,
Periyar University, Salem.

ABSTRACT:

Designing web crawler to meet the web user requirement is most important where the user interest changes every day and they look for the most precise information at rapid time. There are many designs has been discussed earlier but suffers with the problem of generating précised and relevant result in rapid time. The problem of time complexity and accuracy is based on the indexing approach being used; in this paper a novel multi model dynamic web crawling model is discussed to improve the efficiency of web search. The paper discusses about three different models of similarity indexing namely topical, semantic, and taxonomical. With the computed different similarity measures, the document or the web page is identified as belongs to a class and the method extends the level of indexing into infinitive by performing hierarchical cluster. The method identifies the subspace of any web document in the cluster, based on the above computed similarity measure at each level. The pages of results generated is re-ranked using T^2S (Topical-Taxonomical Semantic) similarity measure. The proposed multi model approach produces efficient results in web search and reduces the time complexity and increases the relevancy of results.

Index Terms:

Web Search, Web Crawling, Hierarchical Subspace Clustering, Semantic Ontology, Taxonomy, and Word net.

1. INTRODUCTION:

The World Wide Web becomes more popular in last decade and the usage of web has become an unavoidable one in modern lifestyle. The people use the web for everything and a part of people spend their most time in the web. The most of the banking transactions and commercial transactions uses the web and the business world runs over the web only. Even though, the people are using the web in their most situation, the user may not know about everything and if the user does not know anything about one thing, then they surf the internet to learn about that.

To perform web search the user submits the input query to the search engine through the web interface and the search engine returns set of web links of the web pages. The result returned by the web search engine and the web link present in the web page is based on the kind of indexing the search engine has used. The most of the search engines, index the web pages according to the meta data of the page, so when a query is submitted then the search engine looks for the query text to be appear in the meta data indexed.

In some situations, the meta data may be speak about some topic and it is not necessary that all the content present in the web page must be relevant to the query submitted. Also some search engines indexes the web links according to the context and returns result to the web user. But the results produced will not be relevant and there will be more irrelevant results and reduces the efficiency of web search. This makes the web user to spend more time in searching and it increases the search time.

There are models which cluster the web pages based on the topic of the web page, but the topic is identified based on the terms present in the web page. In case of taxonomical model, the search engine indexes the web page based the terms frequency of the particular class. In semantic approaches, the indexing is performed based on semantic similarity measure which is computed according to classes available and the properties and relations.

Generally a single cluster is the collection of metadata of more web pages, the meta data may speak about many categories and there will be number of sub divisions and sub classes under any category. For example, the fruit class can be classified as follows:

/ Sappoto

Fruits/Seasonal/Mango/Panganapalli

/Malgova/

/Apple/

/ Non Seasonal/Orange

/Banana/

/

Similarly the web pages can be classified into number of categories and each may have their own sub classes and there will be N level of indexing can be done.

For any web search engine, the hierarchical clustering would be an effective approach and the result produced based on this kind of clustering approach will be more relevant and informatics.

The web crawler is the functional component or model, which submits the user query to the search engine and retrieves the results from the search engines. From the retrieved results, the method, generates the indexing or clustering of web documents references or metadata. Further when the user submits the query relevant to the available indexed results the user will be provided with the available results from the category.

How the topic or semantic or taxonomical feature of any web page is identified is by preprocessing the web documents or the meta data of the web page extracted from the search engine. The extracted texts are parsed and the stop words are removed then the method applies stemming process. The stemmed terms are then tagged to produce unique terms of the web page. Based on extracted features of the web page, the concern method will compute similarity of terms between the pages present in the cluster. Using such taxonomy, i.e a collection of terms towards different category of meanings, may produce efficient results while considering term based approaches but when we talk about semantic meanings of documents, the clustering approach has to consider many factors.

2. RELATED WORKS:

There are different methods has been discussed for the development of web search and web crawling and we discuss few of them here in this section.

Extensive research activities are recently directed towards the Semantic Web as a future form of the Web. Consequently, Web search as the key technology of the Web is evolving towards some novel form of Semantic Web search. A very promising recent such approach is based on combining standard Web pages and search queries with ontological background knowledge, and using standard Web search engines as the main inference motor of Semantic Web search. In, Semantic Web Search and Inductive Reasoning [1], they further enhance this approach to Semantic Web search by the use of inductive reasoning techniques. This adds especially the important ability to handle inconsistencies, noise, and incompleteness, which are all very likely to occur in distributed and heterogeneous environments, such as the Web. We report on a prototype implementation of the new approach and experimental results.

Natural Language Interfaces to Ontologies: Combining Syntactic Analysis and Ontology-Based Lookup through the User Interaction [2], present our system FREyA, which combines syntactic parsing with the knowledge encoded in ontologies in order to reduce the customisation effort. If the system fails to automatically derive an answer, it will generate clarification dialogs for the user. The user's selections are saved and used for training the system in order to improve its performance over time. FREyA is evaluated using Mooney Geoquery dataset with very high precision and recall.

Semantic Web search based on ontological conjunctive queries [3], present a novel approach to Semantic Web search, which is based on ontological conjunctive queries, and which combines standard Web search with ontological background knowledge, as it is, e.g., available in Semantic Web repositories. We show how standard Web search engines can be used as the main inference motor for processing ontology-based semantic search queries on the Web. We develop the formal model behind this approach and also provide an implementation in desktop search. Furthermore, we report on extensive experimental results.

Extracting Knowledge from Web Search Engine Using Wikipedia [5], focus on the problem of determining different thematic groups on web search engine results that existing web

search engines provide. We propose a novel system that exploits semantic entities of Wikipedia for grouping the result set in different topic groups, according to the various meanings of the provided query. The proposed method utilizes a number of semantic annotation techniques using Knowledge Bases, like Wordnet and Wikipedia, in order to perceive the different senses of each query term. Finally, the method annotates the extracted topics using information derived from clusters which in following are presented to the end user.

Extracting Knowledge from Web Search Engine Results [6], focus on the problem of determining different thematic groups on web search engine results that existing web search engines provide. We propose a novel system that exploits a set of reformulation strategies so as to help users gain more relevant results to their desired query. It additionally tries to discover among the result set different topic groups, according to the various meanings of the provided query. The proposed method utilizes a number of semantic annotation techniques using Knowledge Bases, like Word Net and Wikipedia, in order to perceive the different senses of each query term. Finally, the method annotates the extracted topics using information derived from the clusters and presents them to the end user.

SemCrawl: Framework for Crawling Ontology Annotated Web Documents for Intelligent Information Retrieval [12], discusses the conceptual differences between the traditional web and semantic web, specifying the need for crawling semantic web documents. In this paper a framework is proposed for crawling the ontologies/semantic web documents. The proposed framework is implemented and validated on different collection of web pages. This system has features of extracting heterogeneous documents from the web, filtering the ontology annotated web pages and extracting triples from them which supports better inferential capability.

Domain adaptation of statistical machine translation with domain-focused web crawling [13], present a strategy for using such resources depending on their availability and quantity supported by results of a large-scale evaluation carried out for the domains of environment and labour legislation, two language pairs (English–French and English–Greek) and in both directions: into and from English. In general, machine translation systems trained and tuned on a general domain perform poorly on specific domains and we show that such systems can be adapted successfully by retuning model parameters using small amounts of parallel in-domain data, and may be further improved by using additional monolingual and parallel training data for adaptation of

language and translation models. The average observed improvement in BLEU achieved is substantial at 15.30 points absolute.

All the above discussed approaches have the problem of irrelevant results and higher time complexity. To overcome these, we propose a multi model web crawler with hierarchical sub space clustering using semantic ontology, word net and taxonomy.

3. Multi Model Dynamic Web Crawler with High Dimensional Subspace Clustering:

The multi model dynamic web crawler reads the input query and retrieves the result from the search engine for the input query. From the retrieved result the method identifies the set of links returned, and based on the retrieved links the multi model approach computes the three different similarity measure on both the page content and search query. Based on the computed similarity measures, the exact subspace of the page is identified. The proposed approach has various stage of crawling and indexing the web page. Also the method produces re-ranked results to the user, which is performed based on the T^2S(topical-taxonomical semantic similarity) ranking algorithm. The re-ranked results are return to the user and the indexed data is used to perform search in the next input query.

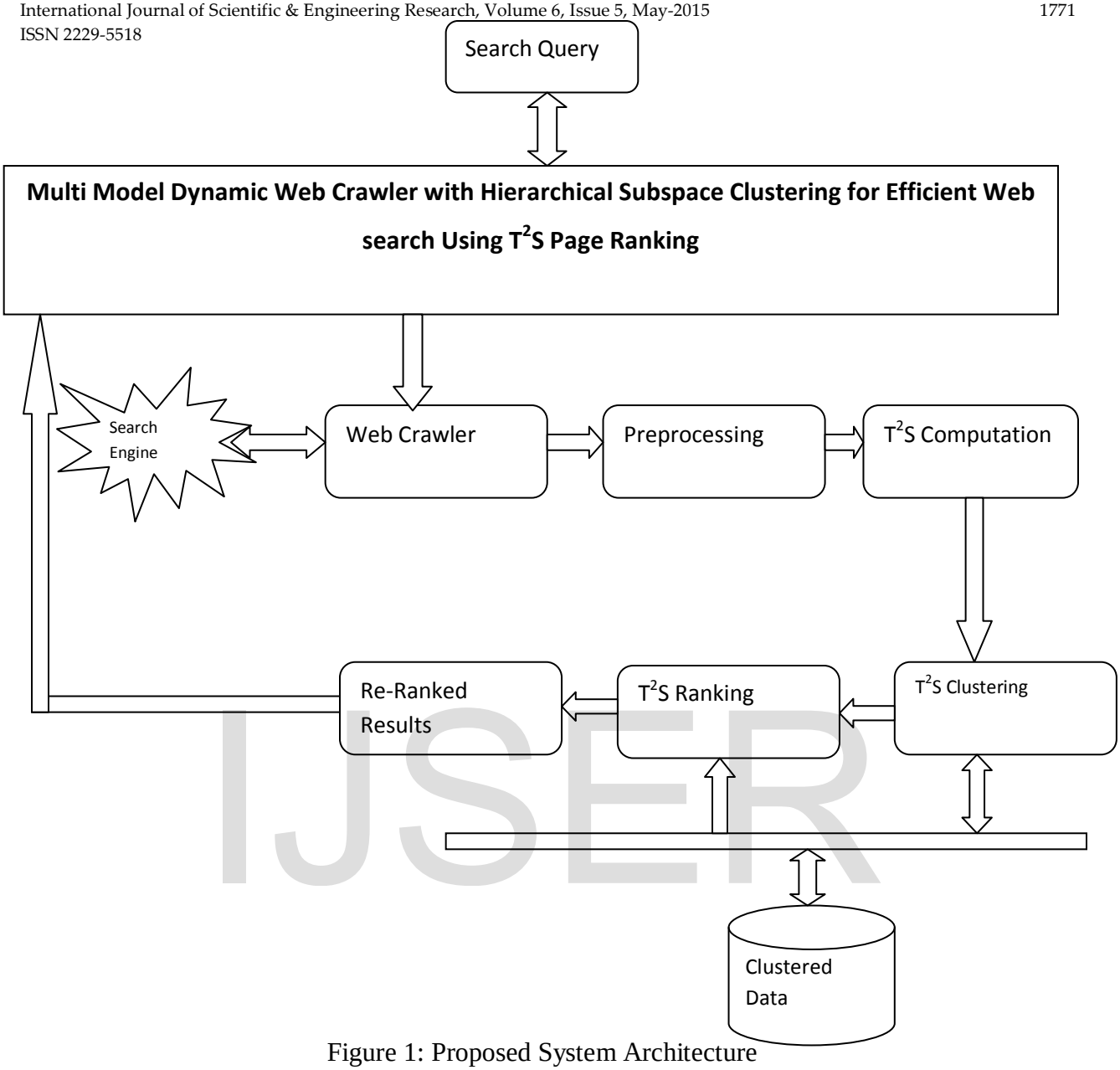


Figure 1: Proposed System Architecture

The Figure 1, shows the overall architecture of the proposed approach and it shows the functional components of the proposed system.

3.1 Web Crawler:

The web crawler is responsible for the query execution and it accepts the input text query from the user and generates number of agents to communicate with the available search engines. The agents submits the query to the search engine and retrieves the results from the search engine. Received results are parsed to a list of web url and the agent performs the lookup in the web link table where there are large set of weblist stored. If the crawler has no such link visited

then the agent reads the page content of the web link and updates the web link table. This modification will be reflected in the web list of other agents and finally the agents produces more number of results. The generated page content are given to the preprocessing stage of the proposed method.

Procedure:

Input: Search Text, Web List wl, Search Engine Set SES.

Output: Text Set Ts.

Start

For each search Engine Si from SES

Generate an Agent and initialize with the query.

SA = $\sum_{k=1}^{size(SES)}$ Create Agent(ST)

Submit query to the search engine.

Receive result Webres = SearchEngine(St).

Parse the links Ls = ∑Links©Webres

For each link Li from Ls

If Wl€Li then

Ignore link.

Else

Wl = ∑li(Wl)+Li //Add the link to the weblink list.

Add to own list

Owl = ∑li(OWl)+Li //Add the link to the weblink list.

End

End.

For each link Li from Owl

Read Page Content Pc.

Add to the text set Ts = ∑(Ts€Ts)+Pc.

End.

End

Stop

The above discussed algorithm, crawls the web links and retrieves the web page content which will be used in the next level of the search process.

3.2 Preprocessing:

At this stage, the method retrieves the page content extracted from the web links and from each of the text, the method replaces the html presentation tags and then with the remaining text, the punctuation marks are removed. Then the method splits the text into single term and adds them into a term set. Now the method performs the stop word removal, stemming and tagging process in the term set. The remaining terms in the term set are the pure nouns and verbs. With the list of nouns and verbs, the method performs the remaining stages of clustering.

Procedure:
Input: Text Set Pc.
Output: Terms set Ts.
Start
For each page content Pc_i from Pc
PC_i = Remove-HTML tage from text.
Term Set Ts = $\sum Terms(Split(PCi, ''))$
for each Term T_i from Ts
if $Ti \in Sl$ then // Sl –stop words list
Ts = TsΠTi
end
end
For each Term T_i From Ts
Perform Stemming.
End
For each term Ti from Ts
Tag t = POS(T_i)
if t==Noun then
else
Remove the term from term set Ts.

End

End

End

Stop.

The above discussed algorithm performs preprocessing of the retrieved page content by removing stop words and stemming each word then tags to identify the pure nouns and verbs from the input text.

3.3 T^2S Computation:

The topical taxonomical semantic similarity measure is compute in this approach. The method uses the wordnet synset pointers to identify the topic of the document and computes topical similarity based on the term frequency of term set present in the document root term set and the set of synset pointers. For each term from the term set of a document, the method identifies the synset pointers, where each of the synset values are looked up for pointers redundant manner. This extracts more meaningful synonyms for the terms of term set. Similarly, the method computes the semantic similarity measure by measuring the class match measure and the number of relations it has within the topic and the number of relation it has with outside the topic. Also the Taxonomical similarity measure is computed according to the level based similarity that the number of terms presents in each level of the taxonomy of concern subspace of taxonomy. By computing all these measures the class and sub space of the document is identified and indexed.

Procedure:

Input: Term set Ts, Taxonomy Tax, Ontology O

Output: Class C.

Start

For each term T_i from Ts

Extract synset from wordnet.

Synset ss = $\sum$Synonyms(Ti)×Wordnet.

Add to Synset Set Syns.

Syns = ∑ss(Syns) U ss.

For each synset point spi from ss

Extract synset pointers.

Synset ssa = ∑Synonyms(spi.Text)×Wordnet.

For each synset point Spa from ssa

If Syns₵spa then

Syns = ∑ss(Syns) U spa.

End

End

End

End

Compute the available classes Css.

Css = ∑Root(Taxonomy)₵Css

For each class Ci from Css

Collect all the terms from the taxonomy.

Term Set Taxt = ∑Terms(Ci)€Tax

Compute Number of terms of Ts present in Taxt.

N = ∑Terms(Ts)€Taxt.

Topical Similarity TopSim = $\frac{N}{Size(Taxt)}$

End

For each Domain Di from O

Compute class match measure Cmm = ∑properties(Di)€Ts

Compute sub class match measure Scmm = ∑properties(Di.properties€Ts

Compute semantic similarity measure Semsim.

Semsim = $\frac{Cmm}{Size(Di.classes)} + \frac{Scmm}{Size(\sum Scmm(Cmm))}$

End

Choose the most top similarity valued topic T = Topic(Max(TopSim))

For each sub class Sci of topic

Compute taxonomical similarity TaxSim = $\frac{\sum Ts(i)€Ts(Sci)}{Size(Ts(sci)}$

End

Compute overall $T^2S = T \times TaxSim$

Choose the most valued sub class Sc with maximum valued T2s class..

The above discussed algorithm computes the topical, semantic and taxonomical similarity measure for each of the domain of class being considered. Based on the topical measure, a single topic or class is identified and then using the semantic and taxonomical similarity values.

3.4 T^2S Clustering:

The clustering algorithm, computes the above mentioned T^2S value for each class and their sub class, then based on the computed measure a single sub class is selected. The web page is indexed into the class being selected.

Procedure:

Input: Web Url

Output: Null

Start

Web Crawler.

Preprocessing

T^2S computation.

Choose the class with most valued T^2S value.

Stop

The above discussed procedure shows the methodology how the clustering is performed by the proposed method.

3.5 T^2S Ranking:

When the clustering is over, and there is a query is preprocessed and with the term set extracted from the query, the method computes topical, semantic and taxonomical similarity measure to identify the class of the query. Once the class of the query is identified then the links already available in the log and the newly obtained results are merged and ranked according to the T^2S measure. The re-ranked result will be given to the user.

4. RESULTS AND DISCUSSION:

The proposed multi model dynamic web crawler with hierarchical sub space clustering approach has been implemented to improve the efficiency of web search and the proposed approach has been implemented and tested for its efficiency. The method has been implemented in java platform and has been validated for various condition. To evaluate the approach the method has been used various search engines like google, Microsoft and many more. The method has produced efficient results in all the cases and has produced efficient search results with more accuracy and relevancy.

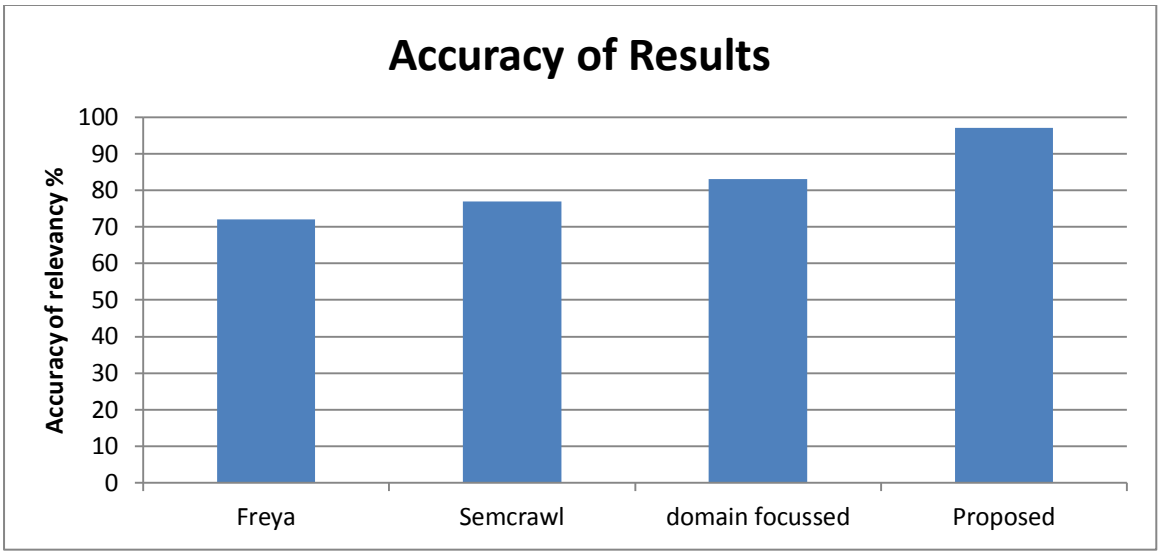
Parameter	Value
Number of search Engines	10
Number of domains	2000
Number of levels	15
Tool used	Advanced Java
Taxonomy	Open Directory Project
Tagger	POS Tagger (Standford)
Lexical Analysis Tool	Wordnet

Table 1: Details of tools used for evaluation

The table 1 shows the details of tools and other parameters and their details used to evaluate the performance of the proposed system.

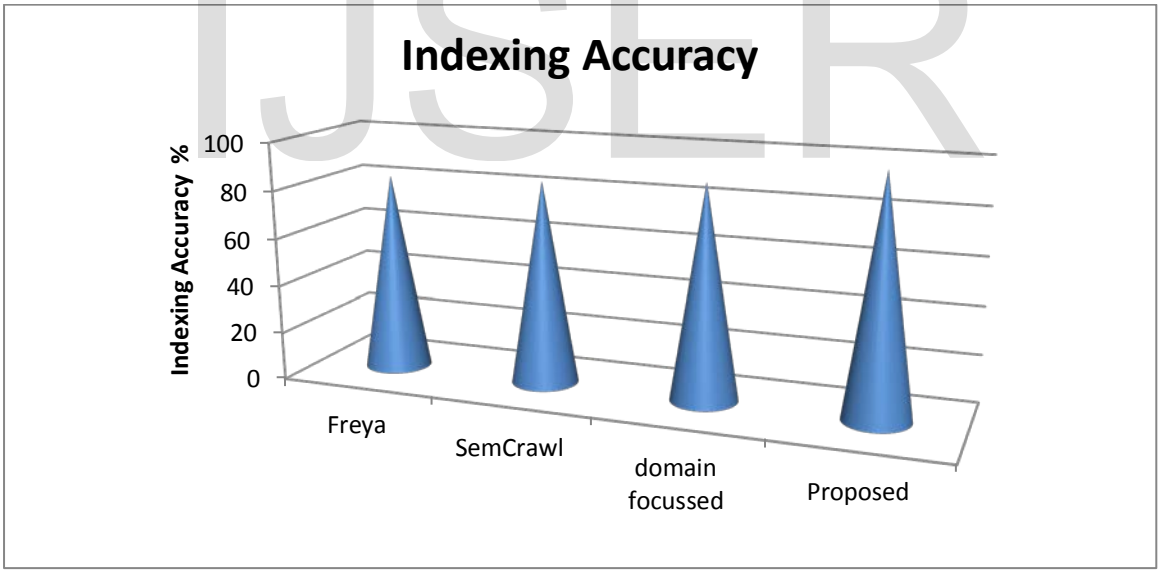
The method considers 2000 number of domains and the basic domain construction is performed using the open directory project (ODP) taxonomy. Because of the system is a semi-supervised learning system, the number of domains will be keep increasing by usage of the proposed method. To tag the input text, the method has used part of speech tagger (POS Tagger) sponsored by standford university.

Also to collect the taxonomy words the method has used word net which gives synset for each of the word i.e similar meaning words to construct the semantic ontology as well as to compute the semantic similarity of the web link.



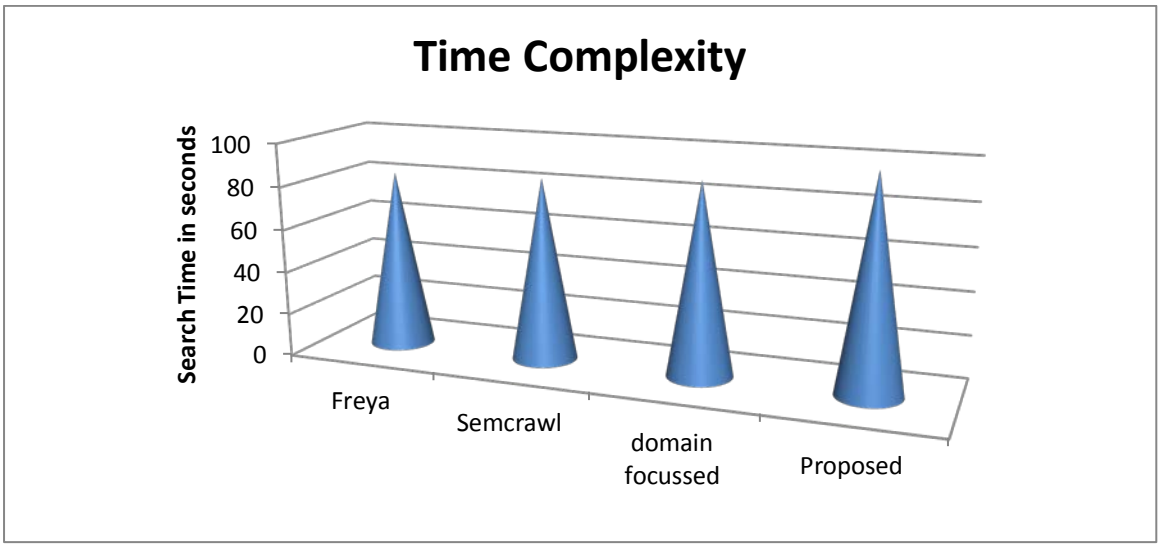
Graph 1: Comparison of relevancy in results

The graph 1, shows the relevancy of results produced by different methods and it shows clearly that the proposed method produces more relevant result than other methods.



Graph 2: Comparison of indexing accuracy

The graph 2, shows the comparative result in indexing accuracy produced by different methods and it shows clearly that the proposed method has produced higher accuracy than other methods.



Graph 3: Comparison of search time complexity

The Graph 3, shows the comparative result in search time produced by different methods and it shows clearly that the proposed method has produced less time complexity than other methods.

5. CONCLUSION:

We proposed multi model dynamic crawler with hierarchical subspace clustering using topical semantically taxonomical similarity measure. The input query is processed through set of search engines and the unique links which are not present in the log are retrieved. The links with their page content are retrieved and preprocessed to remove the stopword, stemming and tagging process. The preprocessed term set is used to compute the T^2S measure. Based on the computed T^2S the document is indexed to a class exactly. To produce the result, the available links with the retrieved results are re-ranked according to the computed similarity measure. Reranked results are given to the user as the final result. The proposed method has produced efficient result in crawling, indexing and producing efficient relevant results. The proposed method has produced less time complexity and reduces the false indexing also.

REFERENCES:

1. Claudia d'Amato, Nicola Fanizzi, Bettina Fazzinga, Georg Gottlob, Thomas Lukasiewicz, Semantic Web Search and Inductive Reasoning, Springer, Uncertainty Reasoning for the Semantic Web II Lecture Notes in Computer Science Volume 7123, 2013, pp 237-261
2. Damljanovic, D., Agatonovic, M., Cunningham, H.: Natural Language Interfaces to Ontologies: Combining Syntactic Analysis and Ontology-Based Lookup through the User Interaction. In: Aroyo, L., Antoniou, G., Hyvönen, E., ten Teije, A., Stuckenschmidt, H., Cabral, L., Tudorache, T. (eds.) ESWC 2010, Part I. LNCS, vol. 6088, pp. 106–120. Springer, Heidelberg (2010)
3. Fazzinga, B., Gianforme, G., Gottlob, G., Lukasiewicz, T.: Semantic Web search based on ontological conjunctive queries. J. Web Sem. 9(4), 453–473 (2011)
4. Janowicz, K., Wilkes, M.: SIM-DLA: A Novel Semantic Similarity Measure for Description Logics Reducing Inter-concept to Inter-instance Similarity. In: Aroyo, L., Traverso, P., Ciravegna, F., Cimiano, P., Heath, T., Hyvönen, E., Mizoguchi, R., Oren, E., Sabou, M., Simperl, E. (eds.) ESWC 2009. LNCS, vol. 5554, pp. 353–367. Springer, Heidelberg (2009)
5. Andreas Kanavos, Christos Makris, Yannis Plegas, Evangelos Theodoridis, Extracting Knowledge from Web Search Engine Using Wikipedia, Springer, Engineering Applications of Neural Networks Communications in Computer and Information Science Volume 384, 2013, pp 100-109.
6. Kanavos, A., Theodoridis, E., Tsakalidis, A.: Extracting Knowledge from Web Search Engine Results. In: ICTAI 2012, pp. 860–867 (2012)
7. Makris, C., Plegas, Y., Theodoridis, E.: Improved text annotation with Wikipedia entities. In: SAC 2013, pp. 288–295 (2013)
8. Scaiella, U., Ferragina, P., Marino, A., Ciaramita, M.: Topical clustering of search results. In: WSDM 2012, pp. 223–232 (2012)
9. Trillo, R., Po, L., Ilarri, S., Bergamaschi, S., Mena, E.: Using semantic techniques to access web data. Inf. Syst. 36(2), 117–133 (2011)

10. Valentin Tablan, Kalina Bontcheva' , Ian Roberts, Hamish Cunningham, Mímir: An open-source semantic search framework for interactive information seeking and discovery, ElseVier, Web Semantics: Science, Services and Agents on the World Wide Web Volume 30, January 2015, Pages 52–68
11. Rishiraj Saha Roy,Rahul Katare[a,] , Niloy Ganguly[a,] , Srivatsan Laxman[b, 1,] , Monojit Choudhury[c], Discovering and understanding word level user intent in Web search queries, Elsevier, Web Semantics: Science, Services and Agents on the World Wide Web Volume 30, January 2015, Pages 22–38.
12. Vandana Dhingra, Komal Kumar Bhatia, SemCrawl: Framework for Crawling Ontology Annotated Web Documents for Intelligent Information Retrieval, Springer, Intelligent distributed computing, vol.321,pp:213-223, 2015.
13. Pavel Pecina, Antonio Toral, Vassilis Papavassiliou, Prokopis Prokopidis, Aleš Tamchyna, Andy Way, Josef van Genabith, Domain adaptation of statistical machine translation with domain-focused web crawling, Springer, Language Resources and Evaluation March 2015, Volume 49, Issue 1, pp 147-193,
14. Banerjee, P., Du, J., Li, B., Naskar, S., Way, A., & van Genabith, J. (2010). Combining multi-domain statistical machine translation models using automatic classifiers. In *Proceedings of the ninth conference of the association for machine translation in the Americas*. Denver, Colorado, USA, pp. 141–150.
15. Hendler, J., Berners-Lee, T. (2010) From Semantic Web to Social Machine. Artificial Intelligence 174: pp. 156-161