AVATAR CRAWLER: FOCUSED WEB CRAWLER FOR ADVANCED SEMANTIC SEARCH ENGINES

R. Aravindhan*   Dr. R. Shanmugalakshmi   K.Ramya*****

**Assistant Professor, Department of CSE, Sri Eshwar College of Engineering, Coimbatore.*
***Associate Professor, Department of CSE, Government College of Technology, Coimbatore.*
****Assistant Professor, Department of CSE, Hindusthan College of Engineering & Technology, Coimbatore.*

Abstract

In the current trend of digitization, a mammoth data subsists in the internet, making it extremely difficult even for the advanced search engine to assuage the desire of the user in extracting exactly what he requires out of the colossal internet space. Providing relevant information to the user has always been a challenging task and it still subsists as a potential research area. Naïve search engines use small programs called crawler that analyses the website and creates an index. The search engine software then sifts through the millions of pages recorded in the index to find the matched pages. One limitation with these unsophisticated search engines is its vulnerability to be deceived by the fake websites that may contain only repeated sentences which will be irrelevant to the user. These search engines fail to identify and filter such sites. A Trade off needs to be achieved between providing adequate information and relevant information. The foresaid crawler just analyses the HTML file and so it is not feasible to track the fake contents. In this paper, a novel search engine based on OWL (Web Ontology Language) file is proposed. The ontology describes the interrelationship among entities in a particular domain. An OWL is written in XML and is designed to be interpreted by computers. It provides a common way to process the content of web information by extracting the semantic relationships between the objects. The proposed ontology based search engine creates an utility crawler that are equipped to automatically generate OWL files from the HTML documents. An OWL file will be generated only when the objects are semantically related and hence it will not be generated for the fake website excluding it from the results provided to the user thereby increasing relevancy. This framework consolidates the technologies of ontology learning and semantic based focused web crawler, in order To enlarge the utilization ratio (the ratio of number of web pages retrieved in first page of the search engine to the number of relevant pages). The innovative unsupervised vocabulary-based ontology learning framework, and hybrid algorithm for matching semantically relevant concepts and metadata is the essence of this research. Array of demonstrations are held to assess the performance of this crawler.

Key Words: Crawler Focused Web Crawler, Ontology Learning, Semantic Tool, Semantic Gap.

Introduction

The World Wide Web (www) is an enormous collection of data. Obviously, the data incorporation increases every milli and micro seconds. Users query is the core that classifies a data’s relevancy and irrelevancy. Researchers focus on techniques that would help others in downloading appropriate web pages. Speaking about the overall view of the researchers, that during search, this huge size of data end up in low revelation of complete data and they also foretell that among the whole, only one third of data has been indexed[1]. Considering the fact, the web is too infinite that the users find the relevant pages over explored as the number of pages retrieved is too high. This scenario leads in downloading the most appropriate and better-defined pages first. As per a conclusion, more than 13% of traffic in the web sites is due to the web search [2]. Apart from other fields, the information technology has a profound effect in the business norms as internet is one of the largest marketplaces in this world. Comparatively, nearly the annual growth has increased to 16% of users been from the year 2001 (360 million users) to 2011 (2 billion users). Search engines are the foundation of the Internet. Most users will turn to a search engine as the quickest way of finding the information, or product that they want. A basic web crawler shall be assumed as a web robot that scans around the web and then downloads the pages that can be retrieved through the links and guides working as an automated program.

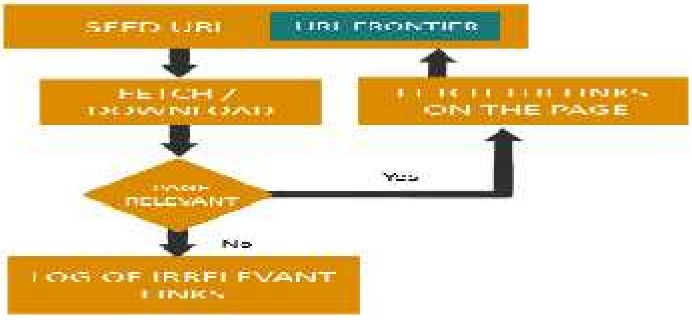
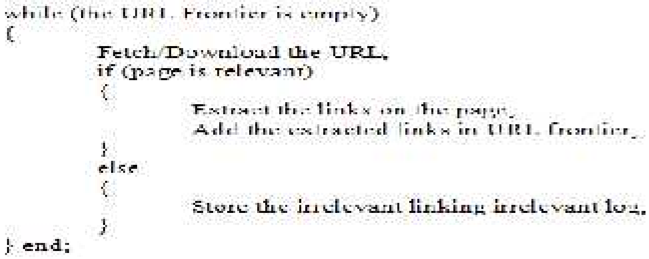


Figure 1: Architecture of Basic Crawler

Initialize the URL Frontier with Seed URL: Fig.1



```
while (the URL Frontier is empty)
{
        Fetch/Download the URL,
        if (page is relevant)
        {
                Extract the links on the page,
                Add the extracted links in URL frontier,
        }
        else
        {
                Store the irrelevant linking irrelevant log,
        }
} end;
```

This process is launched by initializing the URL Frontier in conjunction with seed URL and it continues until and unless the complete URL frontier is completely empty. Each and every URL in the frontier is supposed to be downloaded one after another. Subsequently the relevancy of a page will be fetched. If yes, then obviously the links available in one particular page will be extracted and added to the URL frontier. If no, then the link will be stored in the irrelevant link's log. Assume, that the completion of the whole process will taken place, when the URL frontier becomes empty and the crawler loop terminates; it results in relevant and irrelevant links as an output. Another type of focused crawlers is semantic focused crawler Fig.2 , which makes use of domain ontologies to represent topical maps and link Web pages with relevant ontological concepts for the selection and categorization purposes.[3] In addition, ontologies can be automatically updated in the crawling process. Dong et al.[4] introduced such an ontology-learning-based crawler using support vector machine to update the content of ontological concepts when crawling Web Pages Web search engines deploy the web crawling strategies into two. Viz., Breadth first search strategy and ``Best" first search strategy. Role of the Breadth first search strategy is building a general index in the web that counterparts any feasible topic via attempting to search in a significant portion of the web. Whereas, the ``Best" first search aims in retrieving the page(s) that signifies the given topic. When the crawler uses the ``Best" first search strategy then that particular strategy is stated as ``focused crawler".

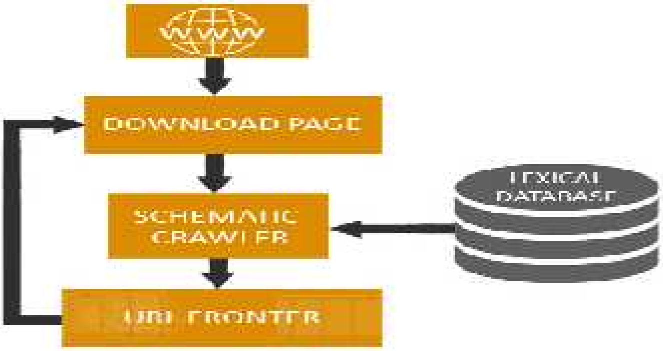


Figure 2: Basic Focused Semantic Crawler

The main components in focused crawlers are: (a) A technique to determine whether a particular page is related to the given topic, and (b) determines the technique to advance from a recognized set of pages. Considering the previous search engine; focused crawling strategy were proposed [5] that relevant pages consist of relevant links. Hence the results were fetched. Whereas, consider of an unfortunate case; there the pages that are not directly connected to the searching stuff are even considered and exposed in the output then that particular search becomes a pre- mature search. This is the main shortcoming. The major difference between a traditional web crawler and the semantic web crawler is that the format given in the source material is traversed and specified to define the link between the information resources. This traditional method operates on HTML; whereas, semantic web crawler operates well on RDF metadata and here links are executed implementing the 'rdfs relationship' [A].

Semantic crawlers are in point of fact the modified version of the classic crawlers. There the priority of page crawling is figured out using the semantic similarity of the web page according to the topic given: that is; a page is related to a topic when they share conceptually similar terms (not necessarily lexically). In ontology [6, 7], the conceptual similarity of terms is defined through the inheritance relationship and others. Each and every condition that is conceptually similar to the terms avail the topic are retrieved from the ontology and used in increasing the depth of the topic.

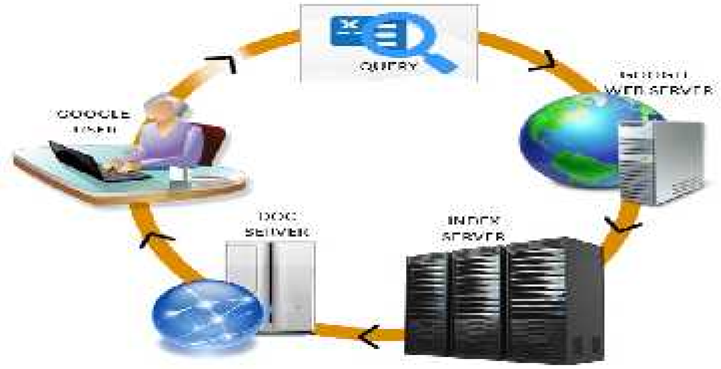


Figure 3: Querying the Semantic Web

Query processing on the Semantic Web [8] is illustrated below in Fig. 3

1. A query enters in with a data type.
2. A server sends queries to the servers' decentralized indexing. The content established on the servers should be parallel in indexing. A book index indicates the pages containing the words matching the query.
3. From there the query travels to the servers where stored documents are retrieved and are generated to describe each search result.
4. The user will be able to retrieve the results of the semantic search that has already been processed in the semantic web server.

Focused Web Crawler [9] is a technique that uses major similarities to map the correspondence between the downloaded page and an unvisited page. Proceeding with this technique promises, only similar pages will be focused and downloaded. Semantic web crawlers uses lexical database in indexing the web pages. Such lexical database affords a help in predicting the associations of the web page and the query entered.

This paper proposes a technique the above two techniques are clubbed altogether (Semantic Focused Web Crawler and Ontology learning) in order to developing a Semantic Based Focused Web Crawler Using Ontology Learning. An ontology learning technology is used as versatile to sustain the higher performance of the crawlers in the Web environment that is totally out of control. The Section II discuss Web Crawler. The section III discusses the Survey of the related works. Section IV deals with proposed framework. Section V converses the proposed methodology with an example. Section VI deliberates the result, section VII contracts with the summary and Section VIII will be concluded with our conclusion and future work.

2. Focused Web Crawler

The Search Engine Spider (or crawler, Robot, SearchBot or Bot) is a program that is used by many of the search engines to find any updating in the Internet. Google's web crawler is termed as GoogleBot. However, there are numerous web spiders in usage. Here we are mainly focusing on Bot because it "crawls" around the web and assembles the documents in order to build a searchable index from the different search engines available. The program starts from a website and follows each and every hyperlink in each page(s).

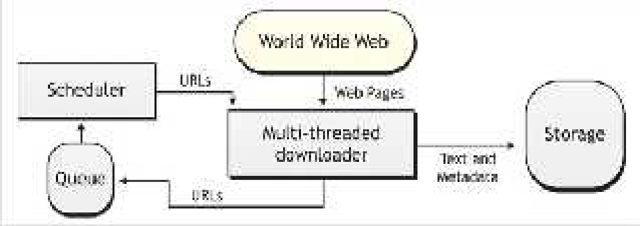


Figure 4: Functional Diagram

Convincingly, here we can pursue and conclude that every query in the web will be originated. Then such "spider" crawls from one website to another website. Evidently, search engines will be simultaneously running thousands of web crawling programs; that too on multiple servers. Figure. 4 Once the web crawler analyses one of your pages; then the same will be loaded into the database as content. Consider, when a page has been fetched, instantaneously the text available in your page will be conjoined into the search engine's index. This is where database is compressed into immense number of words from different web pages. What are we talking about may sound little fishy and technical to many of the people. Anyway, it is a basic step to learn the functionality of the web crawler.

Web crawling procedure features three steps. i) The search bot initializes by crawling into the pages in your web site, ii) indexes the words and content in the site and iii) it migrates into the links (web page addresses or URLs) offered in your site. Imagine, once the spider could not find a page, words will be deleted from the index in a due course. Whereas, there are some other spiders that will check once again to make sure that the link provided has really turned offline. First task done by the spider when it visits your website is checking the file "robots.txt". This file instructs the spider in declaring what are the parts that are to be indexed and what are not to be. This is how the file controls the spider. This is the one of the main rule(s) that every search engine tag on.

Challenges in Web Crawler

Freshness/Coverage Dilemma: Websites such as wikipedia/news sites will be frequently updated and necessitates you to update your crawled content also. You have to bring out a strategy with the intention of roughly estimating the crawl schedule of each domain. You will have to intelligently balance in covering a new domain and updating the covered domains.

Deep Crawling: This is one of the trickiest one that even the internet giants like Google are still trying to find a solution. Calculatingly speaking, the web content hidden behind every HTML forms is of several magnitudes larger than the one that is linked based on the crawl ability of the content in the web. You will have to come up with a smart query generator that is too apt for the particular HTML form. This should also include a size estimator for the underlying data the one that is extracted.

Focused Crawling: Assume, that if your target content is narrower while considering with www, then in such cases you will have to calculate the relevance of the domain keeping in mind about your interests. This results in ensuring you the higher content quality and lower computation/storage overhead. Solution always varies as they range from simple white listing to advanced classifiers.

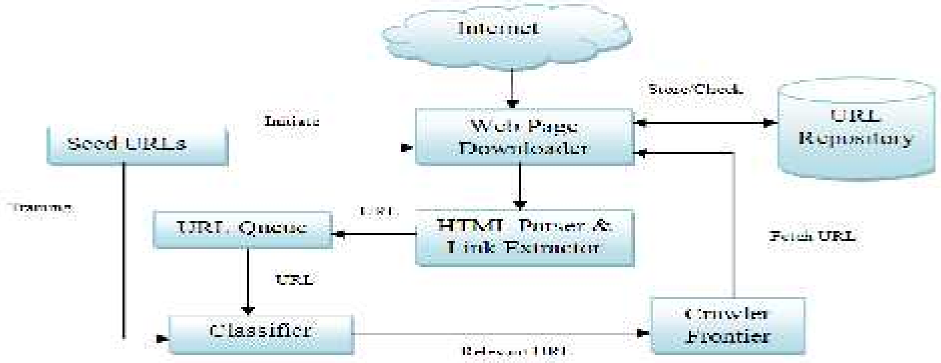
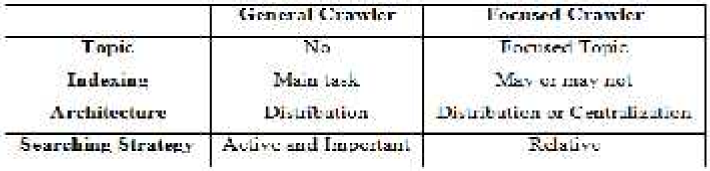


Figure 5: System Architecture of Focused Web Crawler

Table 1: General Crawler Vs Focused Crawler [test]



	General Crawler	Focused Crawler
Topic	No	Focused Topic
Indexing	Main task	May or may not
Architecture	Distribution	Distribution or Centralization
Searching Strategy	Active and Important	Relative

Focused Web Crawler

Focused web crawler is the one that usually searches and retrieves the web pages that are relevant to a specific domain. Figure 1 elaborates the system Architecture of the focused web crawler. The functionality of a perfect crawler is that it should be capable of retrieving the maximum set of relevant pages and at the same time, it should be concurrently traversing the number of irrelevant documents into the minimum possibility on the web [10]. Here, seed URLs are used in initializing the crawling processes. These seed URLs are named as "seed set". Web crawler visits each and every URL(s) available. Then it identifies the various hyperlinks available in the page(s). Different pages will be downloaded from the internet by the parser and thus attained results are stored in a database system of a search engine. URLs then form up as a queue. In every crawling iteration process, the top most link(s) will be chosen and that particular web page will be classified using the classification method. Here the classifications in a web page are relevant and irrelevant. The relevant URL will be added into the crawler frontier. Hence explained process will be continued until and unless the URL queue becomes empty or the limitation of the crawl has been met.

Literature Review

Mejdl S. Safran, Abdullah Althagafi and Dunren Che in 'Improving Relevance Prediction for Focused Web Crawlers'[11] proposes that attaining the relevancy for the unvisited URL as related to the search topic is the main concern while developing a focused web crawler. Effective relevance prediction helps in avoiding the downloading and visiting of too many irrelevant pages. This paper proposes an innovative learning-based approach in means to improvise the relevance prediction in the focused Web crawlers. For this learning, Naïve Bayesian has been used as the basic prediction model, and this can then be later switched to a different type of prediction model. Comparing with the related approaches the experimental result and approach appears to be more valid and efficient in this scenario.

S. Lawrence and C. L. Giles in 'Searching the World Wide Web' [1] while analyzing the coverage and recency of World Wide Web search engines give out some surprising results. [12] However the coverage will be significantly limited in any one of the engines: none of the single engines would have been indexed, more than one-third of the "indexable Web", when we investigate the coverage of six engine they show variation in the order of magnitude. Certainly, the results retrieved as documents from six search engines will be 3.5 times more than the results that accomplished through a single engine. The overlap between pair of engine is analyzed; it outcomes with the lower bound estimation in the size of the indexable web from the 320 million pages.

Carlos Castillo, Mauricio Marin, Andrea Rodriguez in 'Scheduling Algorithms for Web Crawling' [13] proportional study of web crawling strategies are acquainted here. This narrates that the initial practical alternative is combining the breadth-first ordering and largest sites altogether. These results in such a list of advantages as: fast, easy to implement, capable in retrieving the best graded pages and this obviously delivers healthier results than any other preferences. Chilean web crawling is carried over using the simulators. Henceforth, the given strategies are compared amongst same conditions and only actual crawls are processed for validated conclusions.

"Fresh" is a concept that is proposed to maintain the local copies of remote data sources. always the source data is updated independently and parallel. Junghoo et al 'Effective Page Refresh Policies for Web Crawlers' [14] studies about this "Fresh" concept. This is concluded after studying this problem in every search engines. Previously, we were unable to preserve the copies up to date. Through the emergence of this proposal, initially we will have to use two freshness metrics to formalize the notion of "Freshness". In data sources Poisson process is used as a change model. Result is: i) one of the best methods to portray the changes in web pages is Poisson process and ii) thus proposed refresh policies significantly improvises the "freshness" of data. Even in definite cases, the orders of magnitude are undergone by the authors in improvising the existing policies.

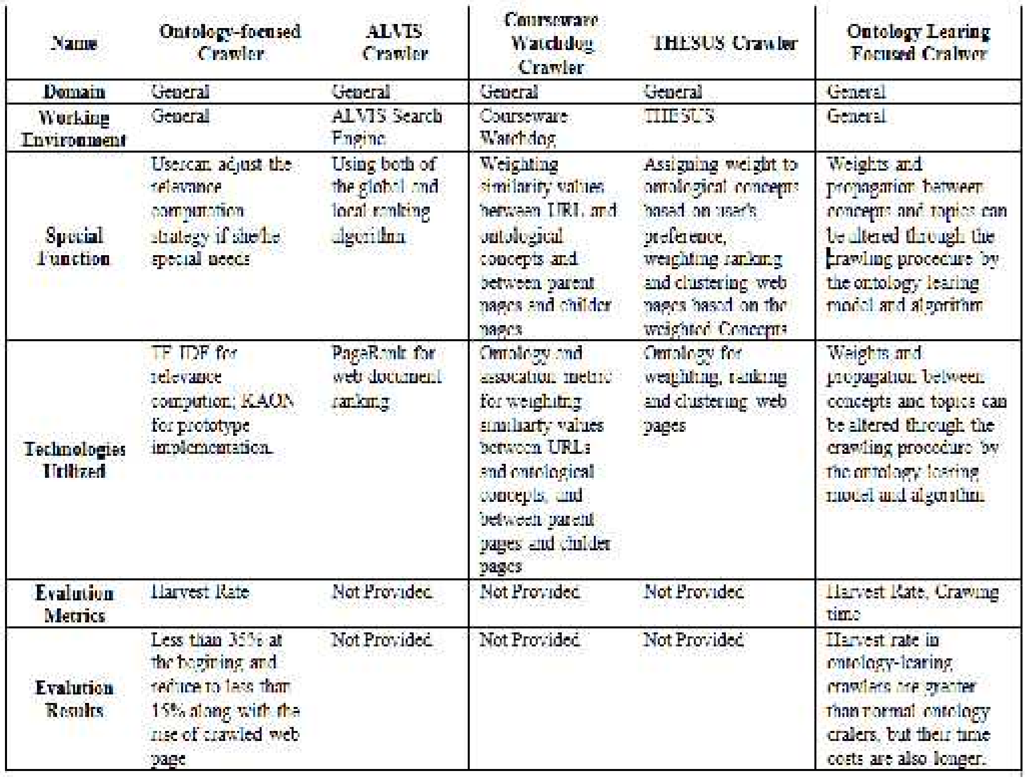
Comparison of the ontology-based focused crawlers In this clause, a brief survey will be taken place now on the ontology-based focused crawlers with six Perspectives special functions, working environment, domain, evaluation metrics, technologies utilized and evaluation results. The contrast result is shown in the below Table.2.

Crawlers shall utilize this multi-domain adaptability feature for the future development reimbursement. The resultant of the comparison table observes that not a bit of the crawlers are domain-specific. Ultimately, the crawlers are allowed to be used in any of the domains and on any crawling topics. The precision can be enhanced and recall will be reduced. This can be achieved by integrating the crawled knowledge using the functionality provided by the crawler. For this, it follows domain-specific heuristic and rules. Considering the working environment, the crawlers can be used as encapsulation in larger systems and also intended as separate tools. Users query topics are always kept on top preference by these. [15,24] Table 2. Comparsion of the Ontology-Based Focused Cralwers crawlers' ontological concepts' functions. This leads to the topic's customization. Here we go towards another crawler that could make it comfortable by evolving the strength between the topics and concepts using an ontology learning model. This has an advanced aspect when compared with the predefined ontology i.e., as it inosculates the crawling topics it would help in deriving the solution to a problem. Conversely, for retrieved web documents ranking the TF-IDF and Page Ranks algorithm are preferably adopted. Apart from the ontology technology, these types of crawlers employ the use of various utilized technologies and those satisfy the different function requirement. In view of the fact that there are no proper crawler evaluation method(s), harvest rate still lend a hand in assessing the crawler performance. Evaluation results will be revealed only in ontology-based focused crawlers and not in traditional web crawlers.

Focused crawling [25-35] has also exploited the Semantic web techniques. Used ontology has been applied by the Semantic focused crawlers for describing the topic of interest and domain. This also rates the performance of the Semantic focused crawler. A domain ontology can be built manually (by domain experts) and automatically (using concepts extraction algorithms). Relevance of the unvisited URLs can be estimated using the ontology built. The actual practical criteria behind this are the comparison done between the concepts in the ontology and concepts extracted from the target web page.

Different fields involved in semantic focused crawling and ontology-learning-based focused crawling are segmented here. The initiative task executed by the ontology learning-based focused crawling is reviewed. According to semantic technologies [36], [37], the semantic focused crawler is in reality a software agent and its main capability relies upon traversing through the web, retrieving and downloading the related web information as per the topic specifies. In the industrial automation [38]–[40], semantic technologies are infinitely functional because; they endow with shared knowledge for improving the interoperability between heterogeneous components. Semantic focused crawler will automatically the semantic underlying in web information as well as in the predefined topics. This directs in retrieving and downloading the relevant web information that too so efficiently and precisely.

Table 2: Comparison of the Ontology-Based Focused Crawlers



Name	Ontology-focused Crawler	ALVIS Crawler	Courseware Watchdog Crawler	THESUS Crawler	Ontology Learing Focused Cralwer
Domain	General	General	General	General	General
Working Environment	General	ALVIS Search Engine	Courseware Watchdog	THESUS	General
Special Function	Usercan adjust the relevance computation strategy if she/he special needs	Using both of the global and local ranking algorithm	Weighting similarity values between URL and ontological concepts and between parent pages and childer pages	Assigning weight to ontological concepts based on user's preference, weighting ranking and clustering web pages based on the weighted Concepts	Weights and propagation between concepts and topics can be altered through the crawling procedure by the ontology learing model and algorithm
Technologies Utilized	TF IDF for relevance compution; KAON for prototype implementation.	PageRank for web document ranking	Ontology and association metric for weighting similarty values between URLs and ontological concepts, and between parent pages and childer pages	Ontology for weighting, ranking and clustering web pages	Weights and propagation between concepts and topics can be altered through the crawling procedure by the ontology learing model and algorithm
Evalution Metrics	Harvest Rate	Not Provided	Not Provided	Not Provided	Harvest Rate, Crawing time
Evalution Results	Less than 35% at the begining and reduce to less than 15% along with the rise of crawled web page	Not Provided	Not Provided	Not Provided	Harvest rate in ontology-learing crawlers are greater than normal ontology cralers, but their time costs are also longer.

Dong et al. [41] survey: Ontology is the main core used by the crawler(s) in this domain to symbolize the web documents and knowledge underlying topics. Obviously, here the performance of the crawler thoroughly depends on the quality of the ontology. We discuss about two issues that affect the quality of ontology. i) We all be acquainted with very well that domain knowledge is accomplished by the ontology using the domain experts. If at all, the domain knowledge achieved by the domain experts varies from the knowledge that of the real world then the worth of the ontology will be affected beyond doubt. ii) Eventually, the knowledge achieved is dynamic and it will surely affect a static ontology. This is contradictory, where the ontology fails in representing the knowledge identical to the real-world knowledge. The versatile and dynamic knowledge and learning capability of humans lead to the short comings in using the semantic based crawler. It becomes the utmost necessity for the researchers to provide a solution on maintaining and enhancing the performance on semantic-focused crawlers and defects in ontologies and they started integrating ontology learning technologies and semantic- focused crawling technologies. Ontology learning techniques are: i) logic-based techniques ii) statistics-based techniques and iii) Linguistics-based techniques etc. from the perspective of learning control, these techniques are further classified into: supervised techniques, semi-supervised techniques, and unsupervised techniques. We have to obtain knowledge from the crawled documents, and integrating it to the ontologies will refine the same. This is declared as a solution to the issue of semantic-focused crawling.

Ontology learning-based semantic focused crawling provides two existing studies and they are reviewed here. Zheng et al. [42] the crawling process involves in maintaining the harvest rate and this is proposed in supervised ontology-learning based focused crawler. The relatedness between the ontology and Web document are determined by artificial neural network (ANN) which is constructed by this crawler. It now becomes the responsibility of the crawler to calculate the term frequency (only relevant concepts) that occurs in the web documents.

Su et al. [43] is proposed to figure out the score on relevance amongst Web documents and topics. A topic represented by a concept in this ontology and specified domain ontology and, the weighted sum is derived from the Web document and the topic’s relevance score of the occurrence frequencies.

Proposed Framework

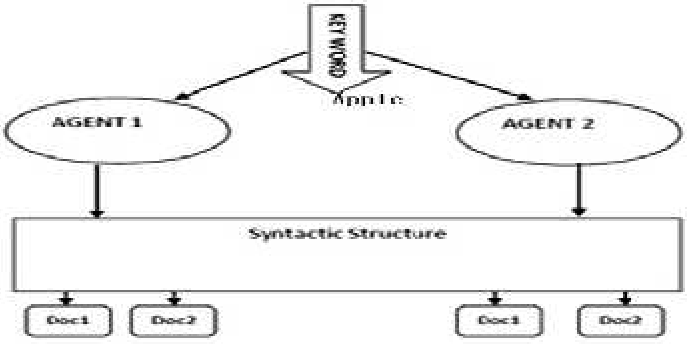


Figure 6: Syntactic Search Results for Keyword ‘Apple’

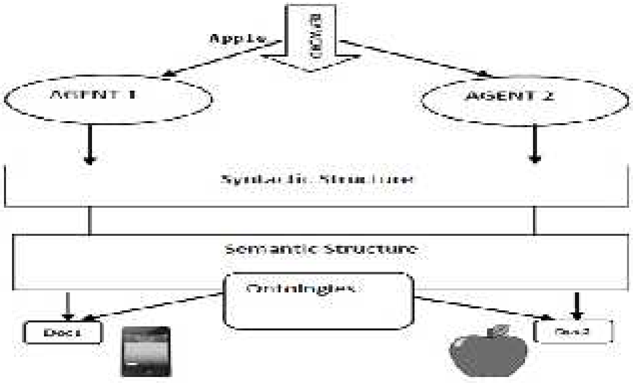


Figure 7: Semantic Search results for Keyword ‘Apple’

In this section, the workflow and the system architecture of the proposed SASF crawler are desperately introduced accordingly. The perceptible tip in this segment is; this exacting crawler is built upon the semantic focused crawler the one which has been designed in our previous search criteria [44]. The differences that are originated in this session and the previous session are summarized below. While taking a view on our previous work, it of course created a pure semantic focused crawler that do not have an ontology-learning function for automatically evolving the utilized ontology. This research aims in creating a remedy for this particular shortcoming.

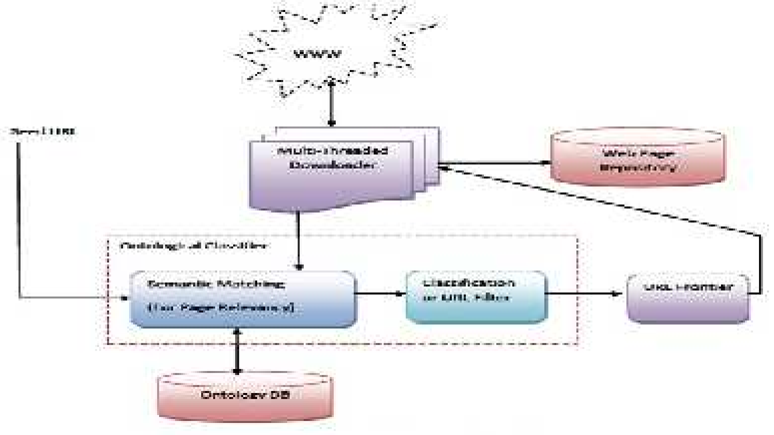


Figure 8: Proposed System Architecture

However we very well knew through the previous section, that the web crawler will be predicting any similarities before downloading a specific web page, therefore it could save the bandwidth of downloading each unvisited page. Conversely, semantic web crawler improvises the quality of the results attained. This paper proposes a technique termed as Semantic Based Focused Web Crawler and it boosts up the accuracy of the semantic web crawler and ensures saving the bandwidth. This functionality has been performed by the focused web crawler when or before downloading a page.

Semantic Similarity Algorithm

The main criteria behind this scenario are, to determine the similarities between a service and concept description. The concept-metadata in semantic similarity algorithm will be elaborated here. This is helpful while tracking the accordance in-between the metadata in the algorithm-based string matching process and the concepts involved (Fig. 1). A hybrid pattern is for gone by this algorithm. That is the aggregation of statistics-based string matching (StSM) and semantic-based string matching (SeSM) algorithm. In the further more section(s), these two algorithms will be feature fully detailed.

A. Semantic-Based String Matching Algorithm

Already it has been elaborated that the SeSM algorithm measures the text similarities between service and concept description. It happens to possible only through a semantic similarity model and WordNet9. Assume that the term processing and preprocessing are complete; then the service and concept description are two different groups of terms. In the First step of approach the semantic similarities are observed. Towards achieving this, we will have take the privilege of using the Resnik [45]'s WordNet and information-theoretic model. From the time when the concepts (or terms) in WordNet are prearranged in a hierarchical structure, and the concepts encloses the relationships from hypernym/hyponym. Via comparing their relative position in WordNet; we will be able to derive the similarities between the two concepts. Resnik's model can be articulated as follows:

$$\text{SimResnik}(C1,C2) = \max_{C€S(C1,C2)}\ [\ -\ \log P(C))]$$

C1 and C2 – Two concepts in WordNet
S(C1,C2) – set of concepts that includes both C1 and C2
P(C) - probability of encountering a sub-concept. Hence,

$$P(C) = \frac{p(C)}{\emptyset}$$

p(C) – number of concepts involved and number of concepts in WordNet.

The noticeable criteria here is that it is not necessarily that the concept should compress of only one term it may or may not possess more than one term. Then the inequity occurs. Resnik's model is within the interval [0, ∞]. We have utilized a model that has been introduced by Dong et al. [23] to normalize the result into the interval that is expressed below:

$$|sim_{\text{Resnik}}(C_1, C_2)| = \begin{cases} \dfrac{\max_{C\in S(C_1,C_2)}[-\log(P(C))]}{\max_{C\in\Theta}[-\log(P(C))]} & \text{if } C_1 \neq C_2 \\ 1 & \text{if } C_1 = C_2 \text{ or } C_1 \in \delta(C_2) \end{cases}$$

Hence, 8 (C2) becomes the synset of C2

In the above step, two terms from service and concept description are used to calculate the similarity values. For instance, suppose the set of vertices are partitioned into two sets. i) P- terms in the service description and terms in the concept description. ii) Q - edge in this graph has an associated weight w within the interval [0, 1] (4).

Plebani et al. [24]'s bipartite graph model is used in assigning the matching optimally.
Graph - G= {V, E}
V – Group of vertices and group of edges that links the vertices.
M E – Matching (two edges in E share a common end vertex)
A task G is a matching M so that incident edge will be there in each vertex.

A function maxSims → [0,1] gives back only the maximum weighted assignment, so the maximum weighted will always be the average weight of the edge. Assignment in the bipartite graph problem is shown as a graphical representation in Fig. 3. Bipartite graph's assignments are be expressed in the linear programming model. Here comes one,

$$\mathrm{maxSim}_S(w, P, Q) = \frac{1}{|P|} \cdot \max \sum_{i \in I}^{j \in J} w(p_i, q_j),$$
$$\forall i \in I, \forall j \in J, I = [1 \ldots |P|], J = [1 \ldots |Q|].$$

B. Statistics-Based String Matching Algorithm

As StSM algorithm does not perform effectively in some of the circumstances it functions in the SeSM algorithm as a complementary solution.Concept description - "mining contractor", Service description - "old mine workings consolidation contractor" Their similarity value is: (1 + 1)/4=0.4 as per the SeSM algorithm. This is credibly lower than the semantic relevance's actual extent. An alternative route has to be defined for measuring similarities, i.e., statistics-based model [25].
(SDi) – Service Description
(CDj,h) – Concept Description

SASF crawler initially downloads the k web pages and then automatically observes the results for the statistical data. From then the semantic relevance will be calculated for the two descriptions. To attain maximum portability for the descriptions (co-occur in web pages), the StSM algorithm follows the unsupervised training paradigm. StSM algorithm is graphically represented in Fig. 4. This could be revealed as follows:

$$\begin{aligned}\mathrm{maxSim}_P(SD_i, CD_{j,h}) &= \max_{CD_{j,\theta} \in C_j} [P(CD_{j,\theta} \mid CD_{j,h}) \cdot P(CD_{j,\theta} \mid SD_i)] \\ &= \max_{CD_{j,\theta} \in C_j} \left[\frac{n_{j,h}^{j,\theta}}{n_{j,h}} \cdot \frac{n_i^{j,\theta}}{n_i} \right]\end{aligned}$$

C. Hybrid Algorithm

Apart from the SeSM and StSM algorithm, a hybrid algorithm is required to search for the maximum similarity values from the two algorithms.

$$\mathrm{maxSim(Sd_i,CD_{j,h})= max[maxSim_s(w_{i,j},Sd_i,CD_{j,h}),naxSim_p(Sd_i,CD_{j,h})]}$$

System Implementation and Evaluation

A. Prototype Implementation

SASF crawler's prototype is implemented in Java platform of Eclipse 3.7.110. We have presented the earlier versions in [11], [12] and prototype is just an extension. In the platform of Protégé 3.4.711, OWL-DL helps in building the Mining Service Metadata Base and Mining Service Ontology Base. The total number of concepts available in service mining ontology is 158. Do you know from where the knowledge is grabbed? It is from: i) Australian Bureau of Statistics13 ii) websites of nearly 200 Australian iii) Wikipedia12 and iv) International mining service companies.

B. Performance Indicators

The comparison between our crawler and the ontology-learning-based focused crawlers will be taken place to define the parameters. In the scenario of ontology-based focused crawling indicators (adopted from IR) are applied only after they are redefined. Harvesting capability of a crawler is calculated using the Harvest Rate. Thus attained harvest rate is concluded below

$$HR(\varepsilon^{\mu}) = \frac{|\sum \alpha^{\mu}|}{|\sum \delta^{\mu}|}$$

Here,
Top – total number of associated metadata commenced from the Web pages.
Down - total number of generated metadata commenced from the Web pages.
Precision - calculates the preciseness of a crawler.

The precision for a concept C j after crawling down the Web pages is determined as follows:

$$P(C_j^{\mu}) = \frac{|\{\alpha_i \in \sum \alpha_j^{\mu} | \alpha_i \in R_j^{\mu}\}|}{|\sum \alpha_j^{\mu}|}$$

Effectiveness of a crawler is calculated using Recall. The recall done on a concept after crawling the Web pages is resolute as follows:

$$R(C_j^{\mu}) = \frac{|\{\alpha_i \in \sum \alpha_j^{\mu} \mid \alpha_i \in R_j^{\mu}\}|}{|R_j^{\mu}|}$$

Aggregated performance of a crawler is measured through Harmonic mean. The harmonic mean (concept C j) once the µWeb pages are crawled is determined as follows:

$$HM(C_j^{\mu}) = \frac{P(C_j^{\mu}) + R(C_j^{\mu})}{P(C_j^{\mu}) \cdot R(C_j^{\mu})}.$$

crawler's inaccuracy is evaluated through the option Fallout. The fallout (concept C j) is firmed after crawling µ Web pages as follows

$$F(C_j^{\mu}) = \frac{|\{\alpha_i \in \sum \alpha_j^{\mu} \mid \alpha_i \in \overline{R}_j^{\mu}\}|}{|\overline{R}_j^{\mu}|}$$

Efficiency of a crawler is calculated using Crawling time. SASF crawlers of a web page's crawling time is defined as per the time interval taken to process the web page through crawling process (Metadata Generation), Association process or to the Filtering process.

C. System Evaluation
Harvest Rate: Fig 9 shows a graphic representation that is yielded through the comparison between increased number of visited web pages and a harvest rate of the ANN, probabilistic and our models. Harvest rate is all about the ability of the crawler and of course not about its accuracy. The harvest rate of the three models is below 60%. On this grounds, in the unlabeled data source, 40% (high propotion) of the web pages are closely viewed as non-mining-service-related Web pages. Here is a strategy in % (percentage) origin of the three models' optimal performance. i) Probabilistic model – 8-9% ii) Our Proposed Model – more than 50% and iii) ANN model – around 16%. Henceforth, our model is on peak of all and it creates a positive impact on the same. This improvises the ability (crawling) of the semantic focused crawler. Constantly, learned concept descriptions matches with the service descriptions that are extracted from many of the web pages.

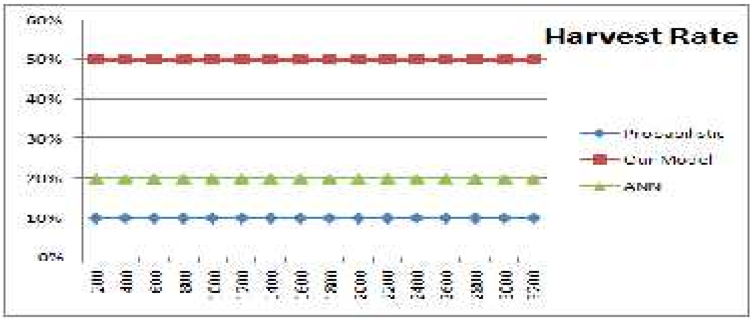


Figure 9: Comparison of the Ontology-Learing-Based Focused Crawling Models on Harvest Rate

Precision: Fig 10 shows a graphic representation which is graphed after the comparison between the number of visited web pages increased and the precision of Our Proposed Model and probabilistic models. The precision of the two models (Self adaptive model and probabilistic model) are 32.50% and 13.46% consecutively. Here, self adaptive model leads. This is for the reason that it filters out many of the irrelevant mining service web pages and to accomplishing the vocabulary-based ontology learning function. From all the above facts, it is well understood that Our Proposed Model model is the only scenario to boost up the semantic focused crawling's preciseness.

Recall: In Fig 11, the graph represents the comparison results of increased number of web pages visited and recall on Our Proposed Model and probabilistic models. The recall percentage of the models is: i) 65.86% (Our Proposed Model) and ii) 9.62% (probabilistic). This model uses the vocabulary-based ontology learning function that ends up in the semantic focused crawler's effectiveness. The answer to the question about the more percentage in the Our Proposed Model model is that it is talented enough to generate mining service metadata relevantly.

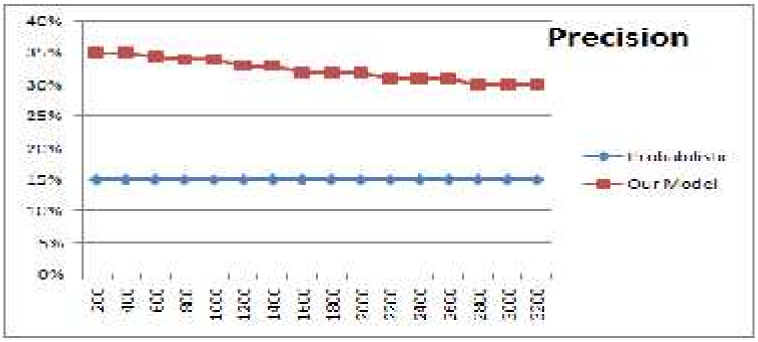


Figure 10: Comparison of the Ontology-Learing-Based Focused Crawling Models on Precision

Harmonic Mean: Fig 12 illustrates a graphical representation that is yielded after the comparison of increased number of web pages visited and the harmonic mean completed on Our Proposed Model and probabilistic models. Due to their least precision performance, the overall harmonic mean values are less than 50% as the parameters are aggregated. The percentage quotient is 43.51% for the Our Proposed Model model and 11.22% for the probabilistic model. Constantly both in recall and precision, the Our Proposed Model is outperforming when compared with probability. The identical resultant is hauled out here with four times the value (%).

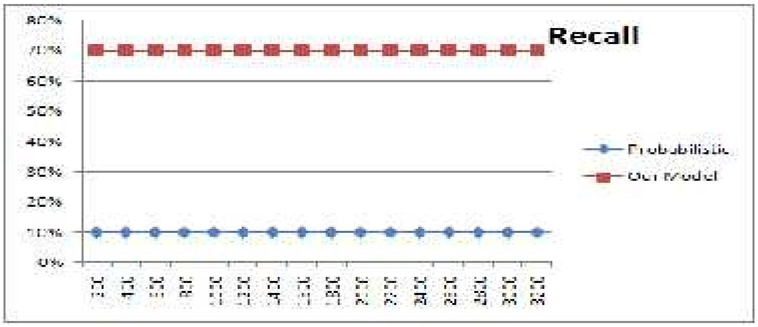


Figure 11: Comparison of the Ontology-Learing-Based Focused Crawling Models on Recall

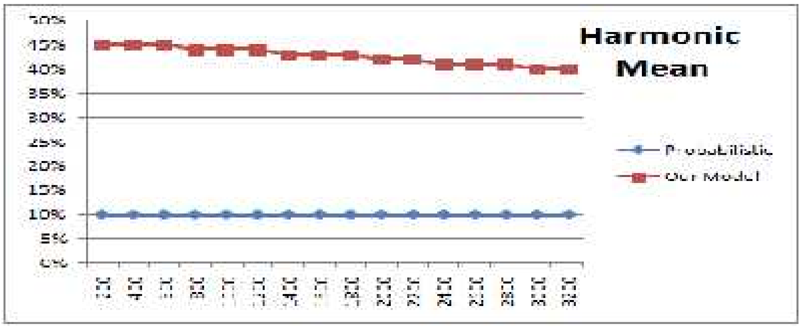


Figure 12: Comparison of the Ontology-Learing-Based Focused Crawling Models on Harmoic Mean

Fig. 13 is representing the graphically expanded output after the comparison prepared between the increased visited web page numbers and the fallout rate on our model and probabilistic models. Obviously, fallout rate calculates the false results. Then there is no wonder that Our Proposed model manage to make less percentage (0.46%) than probabilistic model (0.49%). This scenario definitely concludes that at all times Our Proposed models outperforms with least false results and guarantee their accuracy.

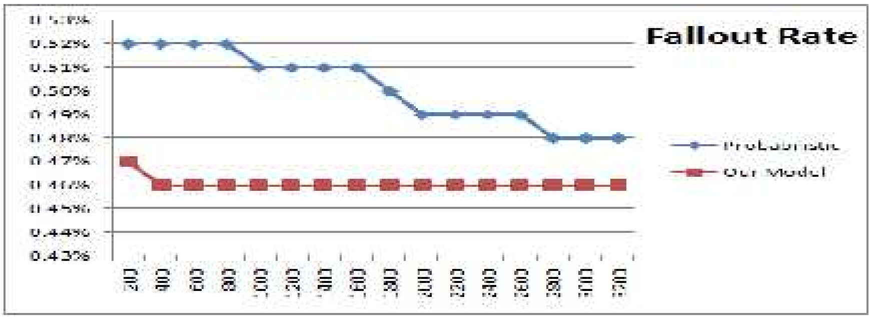


Figure 13: Comparison of the Ontology-Learing-Based Focused Crawling Models on Precision

References

1. S. Lawrence and C. L. Giles, Searching the World Wide Web. Science,280(5360):98.100,1998.
2. StatMarket. Search engine referrals nearly double worldwide.http://websidestory.com/pressroom/pressreleases.html?id=181, 2003.
3. Dong, H., Hussain, F.K., Chang, E.: State of the art in semantic focused crawlers. Computational Science and Its Applications – ICCSA 2009. Springer-Verlag, Seoul, Korea (July 2009) pp. 910-924
4. Jump up^ Dong, H., Hussain, F.K.: SOF: A semi-supervised ontology-learning-based focused crawler. Concurrency and Computation: Practice and Experience. 25(12) (August 2013) pp. 1623-1812
5. T. Berners-Lee, Y. Chen, L. Chilton, D. Connolly, et al. Tabulator: Exploring and analyzing linked data on the Semantic Web. In Proceedings of the ISWC Workshop on Semantic Web User Interaction. 2006.
6. S. Batsakis, E. Petrakis, E. Milios, "Improving the performance of focused web crawlers", Data & Knowledge Engineering, Vol. 68,Issue 10, pp. 1001-1013, 2009.
7. M. Ehrig, A. Maedche, "Ontology-focused crawling of Web documents", Proceedings of the 2003 ACM Symposium on Applied Computing, pp. 1174-1178, USA, 2003
8. Query Processing on the Semantic Web Heiner Stuckenschmidt, Vrije Universiteit Amsterdam
9. Heydon, A., & Najork, M. (1999). Mercator: A scalable, extensible web crawler.World Wide Web, 2(4), 219-229
10. Sunita Rawat, D. R. Patil, “Efficient Focused Crawling based on Best First Search”, 978-1-4673-4529-3/12/$31.00_c 2012 IEEE.
11. Mejdl S. Safran, Abdullah Althagafi and DunrenChe “Improving Relevance Prediction for Focused Web Crawlers”, in the proceding of 2012 IEEE/ACIS 11th International Conference on Computer and Information Science.
12. Pavalam S M, Jawahar M, Felix K Akorli, S V Kashmir Raja “ Web Crawler in Mobile Systems”in the proceedings of International Conference on Machine Learning (ICMLC 2011).
13. Carlos Castillo, Mauricio Marin, Andrea Rodriguez,“Scheduling
14. Algorithms for WebCrawling”in the proceedings of WebMedia and LAWeb, 2004.
15. Junghoo Cho and Hector Garcia-Molina “Effective Page Refresh Policies for Web Crawlers” ACM Transactions on Database Systems, 2003.
16. Ardö, "Focused crawling in the ALVIS semantic search engine," in 2nd European Semantic Web Conference (ESWC 2005), Heraklion, 2005.
17. Y.-J. Chen and V.-W. Soo, "Ontology-based information gathering agents," in Web Intelligence: research and development, N. Z. e. al., Ed. Maebashi: Springer-Verlag, 2001, pp. 423-427.
18. M. Ehrig and A. Maedche, "Ontology-focused crawling of web documents," in ACM Symposium on Applied Computing (SAC 2003), Melbourne, 2003.
19. M. Ehrig, A. Maedche, S. Handschuh, L. Stojanovic, and R. Volz, "Ontology-focused crawling of web documents and RDF-based metadata," in Intenational Semantic Web Conference 2002 (ISWC 2002), Sardinia, 2002.
20. M. Halkidi, B. Nguyen, I. Varlamis, and M. Vazirgiannis, "THESUS: organizing web document collections based on link semantics," The VLDB Journal, vol. 12, pp. 320–332, 2003.
21. G. S. Pedersen, A. Ardö, M. Cromme, M. Taylor, and W. Buntine, "ALVIS - superpeer semantic search engine," in Research and Advanced Technology for Digital Libraries, J. G. e. al., Ed. Alicante: Springer-Verlag, 2006, pp. 461-462.

22. C. Su, Y. Gao, J. Yang, and B. Luo, "An efficient adaptive focused crawler based on ontology learning," in the 5th International Conference on Hybrid Intelligent Systems (HIS'05), Rio de Janeiro, 2005.
23. J. Tane, C. Schmitz, and G. Stumme, "Semantic resource management for the web: an elearning application," in WWW2004, New York, 2004.
24. H. Dong, F. K. Hussain, and E. Chang, "State of the art in semantic focused crawlers," in 2009 IEEE International Conference on Industrial Technology (ICIT 2009), Gippsland, in press.
25. Batzios, C. Dimou, A. L. Symeonidis, and P. A. Mitkas, "BioCrawler: An intelligent crawler for the semantic web," Expert Systems with Applications, vol. In Press, Corrected Proof, p. 908.
26. S. Sizov, S. Siersdorfer, M. Theobald, and G. Weikum, "The BINGO! focused crawler: From bookmarks to archetypes," presented at the Proceedings of the 18th International Conference on Data Engineering San Jose, CA , USA, 2002.
27. Batzios, C. Dimou, A. L. Symeonidis, and P. A. Mitkas, "BioCrawler: An intelligent crawler for the semantic web," Expert Systems with Applications, vol. 35, pp. 524-530, 2008.
28. S. Thenmalar, "Concept based Focused Crawling using Ontology," International Journal of Computer Applications, vol. 26, pp. 29-32, 2011.
29. S. Chang, G. Yang, Y. Jianmei, and L. Bin, "An efficient adaptive focused crawler based on ontology learning," presented at the Proceedings of the 5th International Conference on Hybrid Intelligent Systems (HIS), 2005.
30. Z. Zhang, O. Nasraoui, and R. V. Zwol, "Exploiting Tags and Social Profiles to Improve Focused Crawling," presented at the Proceedings of the IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology -Volume 01 2009.
31. G. Almpanidis, C. Kotropoulos, and I. Pitas, "Focused crawling using latent semantic indexing–An application for vertical search engines," Research and Advanced Technologyfor Digital Libraries, pp. 402-413, 2005.
32. L. Kozanidis, "An Ontology-Based Focused Crawler," in Natural Language and Information Systems. vol. 5039, E. Kapetanios, V. Sugumaran, and M. Spiliopoulou, Eds., ed: Springer Berlin / Heidelberg, 2008, pp. 376-379.
33. S. Ganesh, M. Jayaraj, V. Kalyan, S. Murthy, and G. Aghila, "Ontology-based Web Crawler," presented at the Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC), Las Vegas, Nevada, USA, 2004.
34. M. Ehrig and A. Maedche, "Ontology-focused crawling of Web documents," presented at the Proceedings of the 2003 ACM symposium on applied computing, Melbourne, Florida, 2003.
35. Maedche, M. Ehrig, S. Handschuh, L. Stojanovic, and R. Volz. (2002). Ontology-Focused Crawling of Web Documents and RDF-based Metadata. Available: http://projekte.l3s.uni-hannover.de/pub/bscw.cgi/S4893f6f4/d5269/Maedche_Ehrig-Focused_Crawler-ISWC2002sub.pdf
36. J. J. Jung, "Towards open decision support systems based on semantic focused crawling," Expert Systems with Applications, vol. 36, pp. 3914-3922, 2009.
37. H. Dong and F. K. Hussain, “Focused crawling for automatic service discovery, annotation, and classification in industrial digital ecosystems,” IEEE Trans. Ind. Electron., vol. 58, no. 6, pp. 2106–2116, Jun.2011.
38. H. Dong, F. K. Hussain, and E. Chang, “A framework for discovering and classifying ubiquitous services in digital health ecosystems,” J. Comput. Syst. Sci., vol. 77, pp. 687–704, 2011.
39. J. L. M. Lastra and M. Delamer, “Semantic web services in factory automation: Fundamental insights and research roadmap,” IEEE Trans. Ind. Informat., vol. 2, no. 1, pp. 1–11, Feb. 2006.
40. S. Runde and A. Fay, “Software support for building automation requirements engineering—An application of semanticweb technologies in automation,” IEEE Trans. Ind. Informat., vol. 7, no. 4, pp. 723–730, Nov. 2011.
41. M. Ruta, F. Scioscia, E. Di Sciascio, and G. Loseto, “Semantic-based enhancement of ISO/IEC 14543–3 EIB/KNX standard for building automation,” IEEE Trans. Ind. Informat., vol. 7, no. 4, pp. 731–739, Nov. 2011.
42. H. Dong, F. Hussain, and E. Chang, O. Gervasi, D. Taniar, B. Murgante, A. Lagana, Y. Mun, and M. Gavrilova, Eds., “State of the art in semantic focused crawlers,” in Proc. ICCSA 2009, Berlin, Germany, 2009, vol. 5593, pp. 910–924.
43. H.-T. Zheng, B.-Y. Kang, and H.-G. Kim, “An ontology-based approach to learnable focused crawling,” Inf. Sciences, vol. 178, pp.4512–4522, 2008.
44. C. Su, Y. Gao, J. Yang, and B. Luo, “An efficient adaptive focused crawler based on ontology learning,” in Proc. 5th Int. Conf. Hybrid Intell. Syst. (HIS ’05), Rio de Janeiro, Brazil, 2005, pp. 73–78.
45. Q. Xu and W. Zuo, “First-order focused crawling,” in WWW ‟07: Proceedings of the 16th international conference on World Wide Web, pp. 1159–1160, 2007.
46. P. Resnik, “Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language,” J. Artif. Intell. Res., vol. 11, pp. 95–130, 1999.