JAIDM
Journal of AI and Data Mining
Vol. 2, No. 1, 2014, 25-31.

Prioritizing the ordering of URL queue in focused crawler

D. Koundal

University Institute of Engineering and Technology, Panjab University, Chandigarh, India

Received 14 March 2013; accepted 13 July 2013
**Corresponding author: koundal@gmail.com (D. Koundal)*

Abstract
The enormous growth of the World Wide Web in recent years has made it necessary to perform resource discovery efficiently. For a crawler, it is not a simple task to download the domain specific web pages. This unfocused approach often shows undesired results. Therefore, several new ideas have been proposed, and crawling is a key technique, which is able to crawl particular topical portions of the World Wide Web quickly without having to explore all web pages. Focused crawling is a technique, which is able to crawl particular topics quickly and efficiently without exploring all WebPages. The proposed approach does not only use keywords for the crawl, but also rely on high-level background knowledge with concepts and relations, which are compared with the texts of the searched page.
In this paper, a combined crawling strategy is proposed that integrates the link analysis algorithm with association metric. An approach is followed to find out the relevant pages before the process of crawling and to prioritize the URL queue from downloading higher relevant pages to an optimal level based on domain dependent ontology. This strategy makes use of ontology to estimate the semantic contents of the URL without exploring which in turn strengthen the ordering metric for URL queue and leads to the retrieval of most relevant pages.

Keywords: *WebCrawler, Importance-metrics, Association - metric, Ontology.*

1. Introduction

A crawler is a constituent of search engine that retrieves Web pages by strolling around the Internet following one link to another. A focused crawling algorithm weights a page and extracts the URLs. By rating the URLs, the crawler decides which page to retrieve next. A focused crawler fetches the page that locates on the head of its queue, examines the page and assigns a score to each URL. According to the scores inserted into the queue, the queue will organize itself in order to place URLs with higher scores in the queue head so that they first will be processed. Again, the crawler will fetch the URL on the head of the queue for new processing [1].
Intuitively, the term in-links refers to the hyperlinks pointing to a page. Usually, the larger the number of in-links, the higher a page will be rated. The assumption is made that if two pages are linked to each other, they are likely to be on the same topic. Anchor text can provide a good source of information about a target page, because it signifies how people linking to the page actually describe it. Several studies have tried to use either the anchor text or the text close to it to predict a target page's content. Researchers have developed several link-analysis algorithms over the past few years [2-11]. The most popular link-based Web analysis algorithm includes Page Rank.
A major problem of a focused crawler is to effectively order the links at the crawl frontier so that a maximum number of relevant pages are loaded, while loading only a minimum number of irrelevant pages. This is a challenging task because most of the existing focused crawlers use local search algorithms in Web searching. This may miss a relevant page if there does not exist a chain of hyperlinks that connects one of the seed pages to that relevant page.
The whole paper divides into the following sections: The section 2 discusses the related work

done so far on this challenge. Section 3 gives various prioritizing algorithms. Section 4 tells about association metric based on ontology. Section 5 deals with proposed work on this challenge. The results of experimental evaluation presented in section 6. The implementation details are given in section 7. The section 8 covers conclusion.

2.Related work

Most of the focused crawling techniques use link-structures of the web to improve ordering of URLs in priority queue. A recurring problem in a focused crawling is finding relevant page that is surrounded by non-relevant pages. One remedy presented in [12] by Aggarwal et al. uses the characteristics of the linkage structure of the web while performing the crawl by introducing a concept of "intelligent crawling" where the user can specify an arbitrary predicate (e.g. keywords, document similarity, anything that can be implemented as a function which determines documents relevance to the crawl based on URL and page content) and the system adapts itself in order to maximize the harvest rate. Ehrig et al. in [13] in another approach named as CATYRPEL consider an ontology-based algorithm for page relevance computation. After preprocessing, entities (words occurring in the ontology) are extracted from the page and counted. Relevance of the page with regard to user selected entities of interest is then computed by using several measures on ontology graph (e.g. direct match, taxonomic and more complex relationships). The evaluation of the importance of the page *P* as *I (P)* uses some metrics [14]. Cho et al. proposed an approach calculating the PageRank score on the graph induced by pages downloaded and then using this score as a priority of URLs extracted from a page. This may be due to the fact that the PageRank score is calculated on a very small, non-random subset of the web and also that the PageRank algorithm is too general for use in topic-driven tasks. L. page et al. in [15] proposed an approach for calculating the PageRank score on the graph induced by pages downloaded so far and then using this score as a priority of URLs extracted from a page. They show some improvement over the standard Breadth-first algorithm. Ontology based web crawler [16] estimates the semantic content of the link of the URL in a given set of documents based on the domain dependent ontology, which in turn reinforces the metric that is used for prioritizing the URL queue. The link representing concepts in the ontology knowledge path is given higher priority. However in this work, the content of the page based on the concepts is also used for determining the relevancy of the page. An approach presented by [17] is used to prioritize the ordering of URLs through using association metric along with other importance metric. The rank or relevancy score of the URL is calculated based on the division score with respect to topic keywords available in a division i.e., finding out how many topic keywords there are in a division in which this particular URL exists and calculates the total relevancy of parent page of the relevancy score of the URL page [18]. The maximal set of relevant and quality page is to be retrieved [19].

In this proposed approach, a combination of importance metric and association metric are presented in order to obtain ordering metric for prioritizing the URLs in queue on the basis of syntactic as well as semantic nature of URL.

3. Importance Metric

For a given Webpage *p*, there are different types of importance metrics, which are as follow:

Back link Count

$I(p)$ is the number of links to page *p* that seem over the entire Web. Intuitively, a page *p* that is linked by many pages is more important than one that is rarely referenced. This type of "citation count" has been used widely to evaluate the impact of published papers.

Page Rank

Page Rank is the connectivity-based page quality metric suggested by Page et al. [15]. It is a static measure to rank pages in the absence of any queries. That is, PageRank computes the "global worth" of each page. Intuitively, the Page Rank measure of a page is similar to its in-degree, which is a possible measure of the significance of a page. The PageRank of a page will be high, if many pages with a high PageRank have links to it, and a page having few outgoing links contributes more weight to the pages, it links to a page containing many outgoing links. Thus, a link from the Yahoo home page counts the same as a link from some individual's home page. However, since the Yahoo home page is more important (it has a much higher *IB* count), it would make sense to value that link more highly. The weighted back link count of page *p* is given by

$$IR(p)=(1-d)+d[IR(t1)/c1+....+IR(tn)/cn]$$

4.Association metric with Ontology

Ontology serves as metadata schemas, providing a controlled vocabulary of concepts, each with unambiguously defined and machine-process able

semantics. By defining shared and common domain theories, ontologies help people and machines to communicate succinctly - supporting semantics exchange, not merely syntax.

Ontology is a description (like a formal specification of a program) of the concepts and relationships that can be for an agent or a community of agents. The essential of an ontology is "***is-a"*** hierarchy. The Reference Ontology thus created would have the following associations like "is a", "part of", "has" relationships.

The ***association metric*** for the URL *u* is estimated based on its relevancy with the reference ontology using proper text classification algorithms. Once the page *p* of the URL *u* is downloaded, the association metric for this page *p* is also calculated and preserved, as it will be a parent page for many links to be crawled. *AS(p)* is the same as all links from that page *p* but it utilizes the Web's hyperlink structure to retrieve new pages by traversing links from previously retrieve ones.

Here an ontology-based strategy is taken into account for page relevance computation. After preprocessing, entities (words occurring in the ontology) are extracted from the page and counted and weight of the page is then calculated. With this, a candidate list of Web pages in order of increasing a priority is maintained. In next section, the core elements of proposed work are discussed in detail.

5. Proposed Work

A. System Overview

The focused crawling method consists of two interconnected cycles. The *first cycle* is ontology cycle that defines the crawling target in the form of instantiated ontology. This cycle also presents the output of the crawling process to the user in the form of a document list and proposals for enhancement of the already existing ontology to the user. The *second cycle* comprises the Internet crawler. It intermingles automatically with the data contained on the Web and retrieves them then it connects to the ontology to determine relevance. The relevance computation is used to select relevant documents for the user and to focus on links for the further search for relevant documents and metadata available on the Web. Our proposed focused crawler is based on domain dependent ontology has following components:

All_URLs queue is employed for storing the list of URLs to download.

Metric Module persistently scans through *All_URLs* to make the refinement decision. It

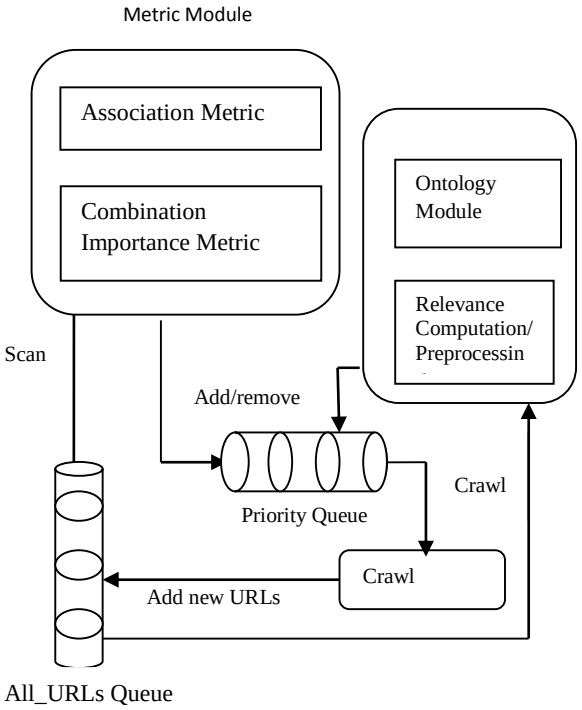

Figure 2. Prototype architecture of ontology based focused crawler

schedules for replacement of the less-important pages in priority queue with the more important page. Metric Module is a collection of Association metric and Combination Metric.

Ontology module works as background knowledge for a crawler to search in the web. It has been widely accepted that ontology is the core ingredient for the Semantic Web. This will have to be extended for the relevance measure of focused crawler. For this purpose, it is a formal and declarative representation, which includes the vocabulary (or names) for referring to the terms in that subject area and the logical statements that describe what the terms are, how they are related to each other, and how they can or cannot be related to each other. Ontology therefore provides a vocabulary for representing and communicating knowledge about some topics and a set of relationships that hold among the terms in the vocabulary. After preprocessing like HTML tag removal, stemming, lexical entries of the ontology are matched against the URLs and a relevance score is computed.

Relevance computation is a function, which tries to map the content (e.g. natural language text, hyperlinks) of a Web document against the accessible ontology to gain an overall relevance-score.

Crawl Module is started with a given set of links. The links are retrieved according to their rank.

Priority queue is used for placing the URLs to be crawled in the front. The URLs in priority queue is chosen by metric module. The processed web resources are indexed and stored in a database and then stored resources are being **s**emantically analyzed and rated in the context of a given ontology. The crawl frontier is implemented by a standard DBMS system.

All crawling modules share the data structures needed for the interaction with the crawler. The prototype maintains a list of unvisited URLs called the frontier. This is initialized with the seed URLs specified at the configuration file. Besides the frontier, the simulator contains a queue. The scheduling algorithm fills it with the first *k* URLs of the frontier, where *k* is the size of the queue mentioned above, once the scheduling algorithm has been applied to the frontier. Each crawling loop involves picking the next URL from the queue, fetching the page corresponding to the URL from the local database that simulates the Web and determining whether the page is relevant or not. If the page is not in the database, the simulation tool can fetch this page from the real Web and store it into the local repository. If the page is relevant, the outgoing links of this page are extracted and added to the frontier, as long as they are not already in it. The crawling process stops once a certain end condition is fulfilled, usually when a certain number of pages have been crawled or when the simulator is ready to crawl another page and the frontier is empty. If the queue is empty, the scheduling algorithm is applied and fills the queue with the first k URLs of the frontier, as long as the frontier contains k URLs. If the frontier doesn't contain k URLs, the queue is filled with all the URLs of the frontier.

B. Proposed Prioritizing Algorithm:

The proposed crawler will work according to the following segment of code.

Input: seed URLs: *start_urls*

Assumption: Initially form beginning assumes Priority queue is full.

***Output*:** Replacing "less important" pages with "more important pages" in a priority queue based on domain specific ontology.

enqueue (url_queue, start_urls);
While (not empty (url_queue) and not termination)
{
url = **dequeue** (url_queue);
page = crawl_page (url);
enqueue (crawled_pages, (url, page));
url_list = extract_urls (page);
For each page p in crawled_pages
Association_weight_page = *AS(p);* // *compute association weight (metric) of page*
End loop

For each u in url_list
enqueue (links, (url, u));
If [u not in url_queue] and [(u ,-) not in crawled_pages]
enqueue (url_queue, u);
Association_Weight_URL = *AS(u);* //*compute association weight of URL*
Combination_Importance = *CI(u);* //*CI(u)= pagerank[u]+ backlink[p]*
End loop

Ordering_metric = O (u);
//

$$O[u] = b_1 CI(u) + b_2 AS(u) + b_3[AS(p_1) + AS(p_2) + ... + AS(p_n)] + b_4 TD[u]$$

where p1, p2 …pn are the parent pages to this url u

reorder_queue (url_queue); //*based on O[u]*

}

C. Ordering Metric O (u)

The ordering metric *O* is used by the crawler for this selection, i.e., it selects the URL *u* such that *O*(*u*) has the highest value among all URLs in the queue. In our experiments, we explore the types of ordering metrics that are best suited for either *IB* (*p*) or *IR* (*p*). The Ordering Metric O(u) used for reordering the URL queue in our crawler is a composite metric defined as follows:

CI (u) = Page Rank[u]

$$O[u] = b_1 CI(u) + b_2 AS(u) + b_3[AS(p_1) + AS(p_2) + ... + AS(p_n)] + b_4 TD[u]$$

Where, p_i is the i_{th} Parent page of URL u to be crawled and b_1, $b2$, b_3, b_4 are real constants to be evaluated from the results of our crawl.

The proposed new ordering metric will solve the major problem of finding the relevancy of the pages before the process of crawling, as well as plays an important role in estimating the relevancy of the links in the page to an optimal level.

6. Implementation details

The implementation of our ontology embedded crawler is an application with in the KAON, the Karlsruhe Ontology and Semantic Web tool suite. The underlying data structure is provided by KAON-API. The crawler is designed with the TextToOnto tool i.e. KAON Workbench. The tight integration of the crawler with the ontology and metadata management component is also important to allow for quick adaption and extension of the

structures. The proposed framework for focused crawling has been implemented in KAON framework and is written in Java.

7. Experimental Results

The results of this paper are the relevant web pages obtained from crawled pages for the different three seed URLs. The resulting comparison charts are drawn using Microsoft Excel software. Graphical interpretations of these results are also shown here.

Performance Metrics

In order to evaluate the performance of a given scheduling algorithm, the metric used is:

Harvest rate

Harvest rate is a common measure on how well a focused crawler performs. It is expressed as

HR= r/p,

Where,

HR is the harvest rate,

r is the number of relevant pages found and

p is the number of pages downloaded.

Seed URLs

For the crawler to start crawling we provide some seed URLs.

http://www.puchd.ac.in (Panjab University),
http://www.du.ac.in (Delhi university),
http://www.ignou.ac.in/ (Indra Gandhi National Open University).

Scenario

1. **http://www.puchd.ac.in/**

In first experimental run, total 1000 pages were crawled from which 478 relevant pages were obtained. Therefore, the harvest ratio obtained for this crawler run is 48%. The harvest ratio for seed URL **http://www.puchd.ac.in:80/** is shown in Figure 4.

From first crawler run, the sample of top ten URLs of

obtained results set is shown in Table 1 as:

2. **http://www.du.ac.in/**

In second experimental run, 464 relevant pages were obtained from total crawled pages i.e. 1,000.

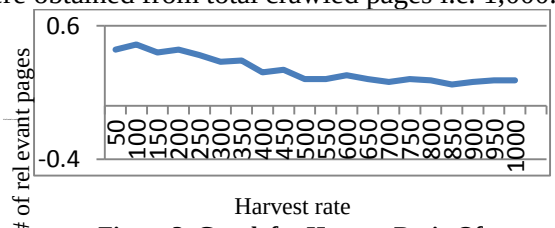

Figure 3. Graph for Harvest Ratio Of http://www.puchd.ac.in/

Table 1. Top 10 results for Panjab University

rank	Web Page
1	http://directory.puchd.ac.in:80/
2	http://exams.puchd.ac.in:80/
3	http://uiet.puchd.ac.in:80/
4	http://puchd.ac.in:80/prospectus.php
5	http://punet.puchd.ac.in:80/
6	http://forms.puchd.ac.in:80/
7	http://admissions.puchd.ac.in:80/
8	http://results.puchd.ac.in:80/
9	http://tenders.puchd.ac.in:80/
10	http://alumni.puchd.ac.in:80/

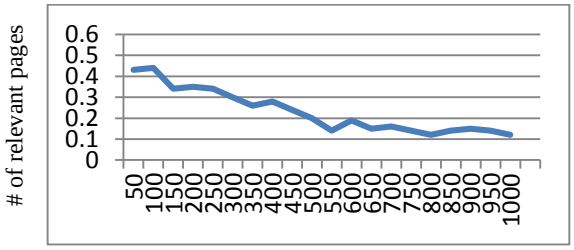

Figure 5. Graph for Harvest Ratio of http://www.du.ac.in/

Therefore, the harvest ratio obtained in this second run is 46% which is shown in Figure 5.

1. **http://www.ignou.ac.in/**

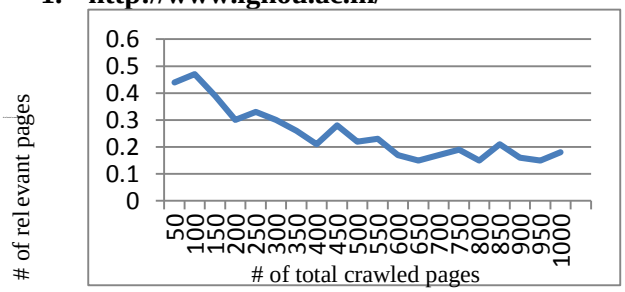

Figure 6. Graph of harvest ratio of http://www.ignou.ac.in/

In the third experimental run, 496 relevant pages are obtained from 1000 crawled pages. Therefore, the harvest ratio obtained in this third run is .49% as shown in Figure 6.

A. Average Harvest Rate Of Three Experimental Run

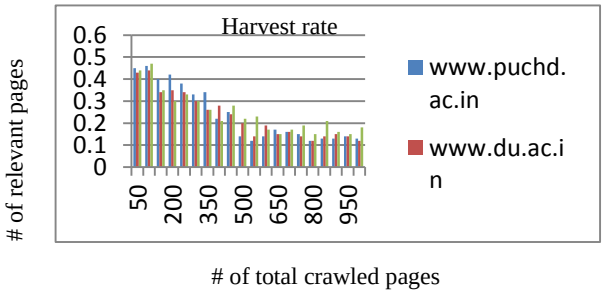

Figure 7. Average Harvest ratio of above three URLs

In above three experimental runs, total 3,000 webpages were crawled from which total of 1,434 pages were obtained. The above results of these three seed URLs i.e www.puchd.ac.in/, www.du.ac.in/, www.ignou.ac.in/ show that our ontology-based focused crawler is better than standard crawler and having average harvest ratio of 48%.

B.Comparison Of Unfocused Crawler And Ontology-Based Crawler:

The literature analysis shows that unfocused crawler with link analysis algorithm crawled 350 pages out of 1000 pages i.e. the obtained harvest ratio is 35% as shown in Table 2.

Table 2. Simulation results of different algorithm

Strategy	# of pages visited	# of relevant pages visited	Harvest Ratio
Breadth First	1,000	287	28%
PageRank	1,000	350	35%
Ontology based crawler	1,000	478	48%

Another evaluation run shows that more relevant pages were` obtained using ontology-based crawler rather than unfocused crawler is given in Figure 8. With the help of ontology-based crawler using link analysis algorithm, the harvest ratio obtained is 48%, while with unfocused crawler having link analysis algorithm, the harvest ratio obtained is 35%. This shows that more relevant pages can be retrieved by using ontology with our proposed combined strategy.

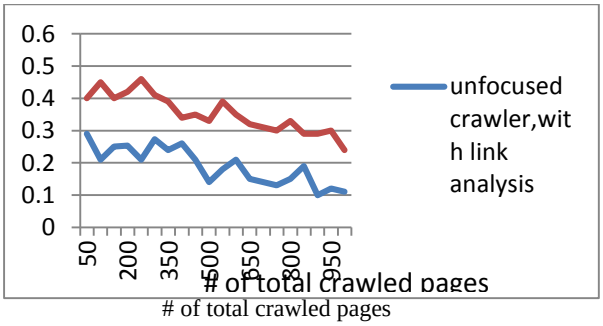

Figure 8. Comparison of unfocused crawler, with link analysis and ontology-based crawler, with link analysis algorithm

8. Conclusion

In this paper, a combined strategy of link analysis algorithm guided by topic ontology is proposed in order to efficiently discover pages relevant to the domain of interest. The prototype uses the structured information in the ontology to guide the crawler in its search for web pages that are relevant to the topic specified in the ontology. The test results show that the use of link analysis in our prototype gives a slight increase in the harvest rate. Our crawler depends on rating the links which in turn enhance the discovery mechanism, with the introduction of combination of importance metric, this distinguishes our approach from existing approaches as the link with the higher calculated rank will be visited next. A final conclusion of this work is the realization that it is definitely worth using advanced knowledge structures when searching a specific domain on the Internet and it is possible to extract much more information from the large distributed database Internet as today's applications allow. This makes it an effective tool for the Semantic Web environment. This may result in improving the performance in the area of focused crawling and overcomes the various drawbacks of the current approaches.

References

[1] Blaz Novak, "A Survey of Focused Web Crawling Algorithms" SIKDD 2004 multi conference IS2004, 12-15 Oct 2004.

[2] Hiep Phuc Luong, Susan Gauch, Qiang Wang, 2009. Ontology-based Focused Crawling, International Conference on Information, Process, and Knowledge Management, pp. 123-128.

[3] Li, H., Peng, Q. Q., Du, Y. J., Zhao, Y., Chen, S. M., Gao, Z. Q. (2009). Focused web crawling strategy based on web semantics analysis and web links analysis. Journal of Computational Information Systems, 5(6), 1793-1800

[4] Batsakisa S, Petrakisa EGM, Milios E. Improving the performance of focused web crawlers. Data Knowl Eng 2009;68(10):1001–13

[5] Chakrabarti S, van den Berg M, Dom B. Focused crawling: a new approach to topic-specific web resource discovery. Comput Netw 1999;31(11–16):1623–40.

[6] Chakrabarti, S., M. Berg, B. Dom, Fo cused crawling: A new approach to topic-sp ecific web resource discovery, Computer Networks and ISDN Systems, 31 (11-16), 1999, 1623-1640

[7] S´anchez, D., M. Batet, D. Isern. Ontology-based information content computation. Knowledge-Based Systems, 24 (2011), 297-303

[8] Mohen Jamali, Hassan Sayyadi, Babak Bagheri, Hariri and Hassan Abolhassani, 2006. A method of focused crawling using combination of link structure and content similarity, Proceedings of the International Conference on Web Intelligence.

[9] Kozanidis L. An ontology-based focused crawler. In: LNCS 5039. Springer; 2008. p. 376–9.

[10] Liu Z, Du Y, Zhao Y. Focused crawler based on domain ontology and FCA. J Inform Comput Sci 2011;8(10):1909–17

[11] Ester M., Gro M. and Kriegel H.-P.: 2001, Focused Web crawling: A generic framework for specifying the user interest and for adaptive crawling strategies, Technical report, Institute for Computer Science, University of Munich.

[12] Aggarwal, C., F. Al-Garawi and P. Yu. "Intelligent Crawling on the World Wide Web with Arbitrary Predicates", In Proceedings of the 10th International WWW Conference, Hong Kong, May 2001.

[13] Ehrig M. and A. Maedche "Ontology-Focused Crawling of Web Documents" Proc. the 2003 ACM symposium on applied computing.

[14] Cho, J., H.Garcia - Molina, and L. Page. Efficient crawling through URL ordering. Computer Networks, 30(17):161172, 1998.

[15] Page, L., S. Brin, R. Motwani, T. Winograd. "The PageRank Citation Ranking: Bringing Order to the Web", Stanford Digital Library Technologies Project.

[16] Ganesh, S., M. Jayaraj, V. Kalyan, and G. Aghila,"Ontology-based Web Crawler," Proc. of the International Conference on Information Technology: Coding and Computing, Las Vegas, NV, USA, pp.337-341, 2004.

[17] Deepika Koundal, Mukesh Kumar, Renu Vig, "Prioritizing the URLs in Ontology based Crawler" published and presented at International Conference of IEEE- AICC '2009 at Thapar University, Patiala.

[18] Debashis Hati, Amritesh kumar, 2010. An approach for identifying URLs based on Division score and link score in focused crawler, International journal of computer applications, Volume 2 – No.3.

[19] Debashis Hati, Amritesh Kumar, Lizashree Mishra, 2010. Unvisited URL Relevancy Calculation in Focused Crawling Based on Naïve Bayesian Classification, International Journal of Computer Applications, Volume 3- No.9.