Synopsis on

DESIGN AND DEVELOPMENT OF A MACHINE LEARNING BASED FOCUSED CRAWLER FOR INFORMATION RETRIEVAL

AUGUST-2019

Submitted for registration in the degree of

Doctor of Philosophy

DEPARTMENT OF CSE

CHITKARA UNIVERSITY

HIMACHAL PRADESH

Submitted by

Ms Shivani Gautam (PHDENG17051)

Under the joint supervision of

Dr. Shaily Jain
Chitkara University,
Himachal Pradesh

Dr. Rajesh Bhatia
Punjab Engineering College
(Deemed University),Chandigarh

Table of Contents

1. Introduction

Today, most of the popular search engines like Google, Bing, and Yahoo etc have provided us with facilities to locate any information on the Internet. When a user tries to search for any information, he usually focuses on some specific topic or person. Search engines use Web crawlers to gather information available online. Web crawlers are the devices that keep on following the hyperlinks to gather information. Rather than collecting all the available data on the Web, focused crawler selectively download web pages that are relevant according to a predefined criteria. The concept of focused crawling was introduced in [1]: a focused crawler can seek, acquire, and index web pages on a specific set of topics that represent a narrow segment of Web. Focused crawling approach leads to important savings in hardware and network resources, and helps to keep the data gathered by the crawler more abreast.

1.1 Search Engine

A search engine can be defined as a program created to find information from the World Wide Web (WWW). The search engine produces a result by searching indexed database as per the user request. Generally, the criteria are specified in terms of keywords or phrases. The results fetched are given in an organized manner that matches the specified criteria. At the back end, search engines use most frequently updated indexes to function quickly and efficiently. Search engines maintain their database index by searching a large portion of Web.

1.2 Web Crawler

A Web crawler is also known as a Web spider or Web robot. These are the automated computer scripts or software that browses the WWW iteratively by following the hyperlinks [2]. The method of retrieving data from Web by a crawler is called web crawling. Web crawlers download the visited web pages so that an index of these web pages can be created. A Web crawler starts with a list of

Uniform Resource Locator (URLs) to visit, called the seed URLs. As the crawler begins, it retrieves all the hyperlinks in the webpage and adds them to a list of URLs to be visited further [3].

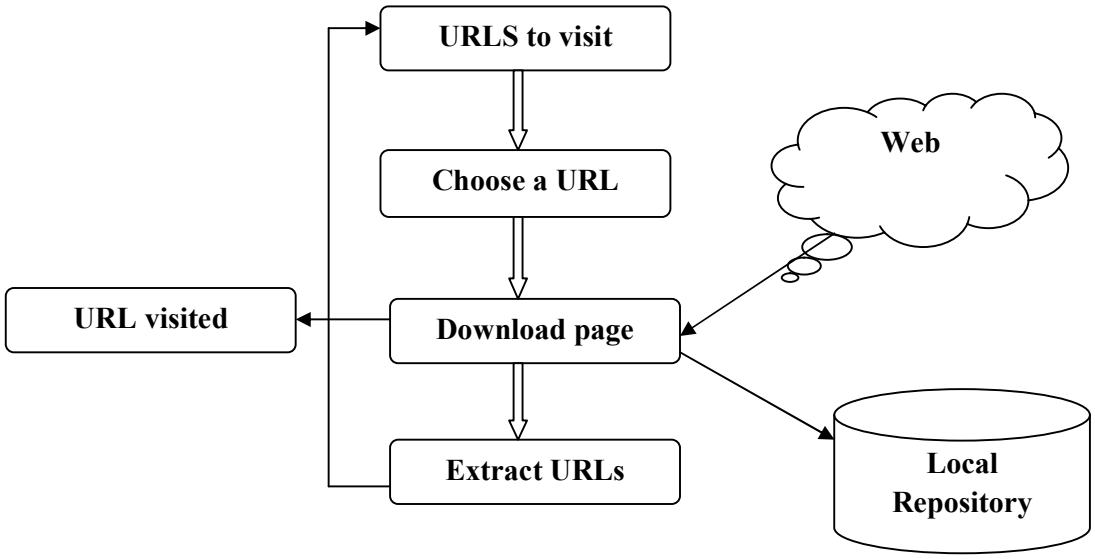

Fig 1: Architecture of a web crawler [3]

The goal of crawling is to collect as many useful web pages as possible in the minimum possible time. Crawler emphasizes the order in which the URLs are visited due to huge collection of data on the web. Huge size of the Web makes it impossible for any crawler to retrieve all the relevant data from the Web. Therefore, various types of Web crawlers have surfaced as an active research area. Web crawlers are the tools for data collection in the search engines. These are also called as spiders or robots or wanderers. A simple basic Web crawler is a function with a set of seed URLs as input and a set of crawled web pages as output. This simple function takes URL one by one, gets the webpage, and adds URLs found on this webpage to the list of URLs to be visited further. Brin and Page [2] discuss the general basic architecture of the Web crawler and the different data structures that can be used. As shown in Figure 1, URL to visit and URL visited so far are preserved to keep track of the web pages visited so far. Single URL is selected from the list of URLs and the corresponding webpage is downloaded at the local site. From the downloaded webpage, URLs are extracted and added to the list of URLs to visit. This basic architecture of focused crawler can be improved according to the requirements. In the architecture, more components can be added, or existing components can be modified as per the requirements. A simple crawler needs at least these components: A set of seed URLs for starting the crawling

process, page downloader to download web pages from the Web to the local storehouse or Web depository, URL extractor to get the URLs from web pages. This method is repeated periodically until the list of URLs to be visited is empty.

1.3 Types of Web Crawler

1. Universal or Broad crawler: This type of Web crawler is not limited to web pages of a specific topic or domain. It keeps on following links repeatedly and gets all web pages they come across.
2. Preferential crawler: This type of Web crawler does not crawl all the links they come in contact with; instead the user gives a specific topic of interest that guides the preferential crawler. Preferential crawler can be classified as **focused and topical crawler**. Chakrabarti et al. [1] proposed one of the very first focused crawler that selectively looks for web pages that are relevant to a predefined set of topics. Topical crawler is used for searching information related to a specific topic from the Web. Topical crawling infers that only the topic of interest is specified whereas focused crawling infers that some labeled examples of relevant and non relevant web pages are also provided [4]. **Forum crawler** deals with crawling the online forum content. It provides the users space to share, discuss and request information.
3. Hidden Web crawler: A huge part of information on the Web cannot be accessed directly by the simple crawlers by following the hyperlinks on web pages. This information is hidden behind search or query interface and can be found out by only submitting queries to database, these web pages lie in hidden Web or deep Web [5]. A special category of crawlers called hidden Web crawler deals with crawling this section of the Web.
4. Mobile crawler: This type of web crawler crawls the data in a way where selection and filtration of web pages are carried out on server side rather than on the search engine side. These crawlers reduce network load caused by universal Web crawlers.[6] [48]

5. Continuous or Incremental crawler: Based on the estimates as to how often the web pages change, this type of crawler refreshes the existing collection of web pages by visiting them frequently thereby keeping the database of the search engine up-to-date [8]. Nevertheless, there is a bargain between managing page freshness and resource utilization.

1.4 Focused Web Crawler

By Definition "a focused crawler can seek, acquire and index web pages on a specific set of topics that represent a narrow segment of web". It tries to find high grade information on a specific topic while avoiding irrelevant links. It downloads selective web pages that are relevant according to predefined benchmark. Focused crawling technique gives priority to those urls in the method of crawling where the chances of finding user specific information is high. It consists of components like classifier and distiller as shown in below diagram. The classifier evaluates the relevance of the page and the distiller identifies the hypertext links that points to relevant pages.

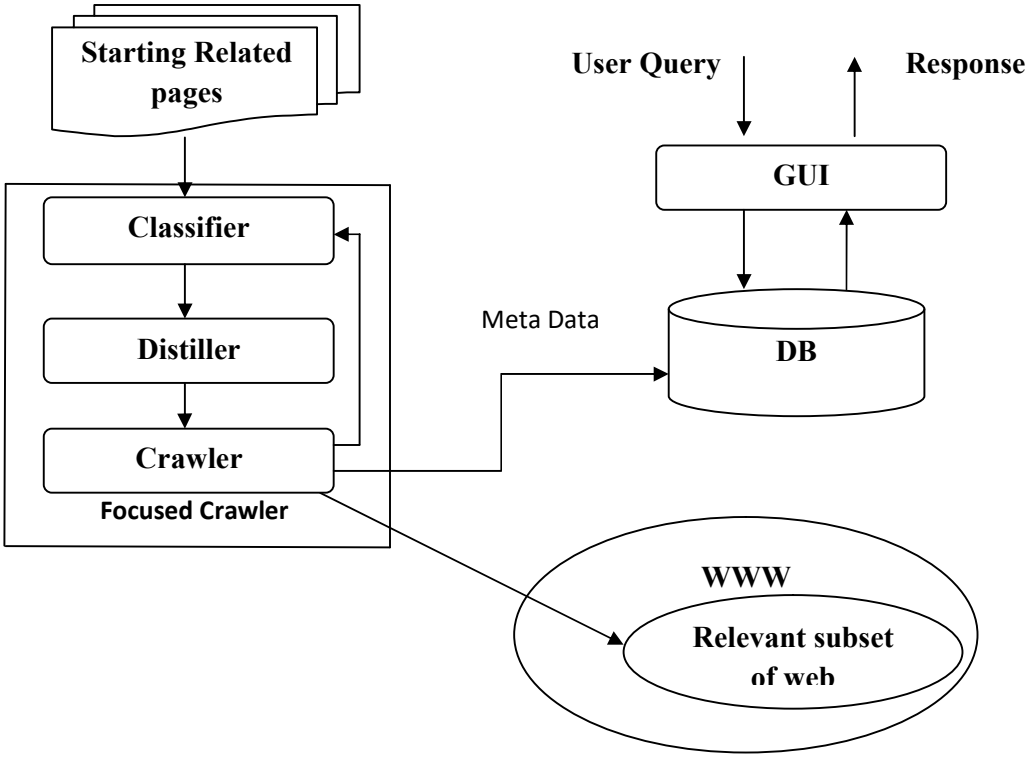

Fig2: Architecture of a Focused crawler [9]

1.5 Subcategories of focused, forum and topical crawler

- Focused crawler using soft computing techniques.
- Application-based focused crawler.
- Focused crawler based on link, text and URL.
- Focused crawler based on context, graph, decision tree, and DOM.
- Semantic crawling-based focused crawler.
- Topic specific focused crawler.
- Parallel and distributed focused crawler.
- Language classification based focused crawler.
- Vertical search engine based on focused crawler.
- Query, keyword, and metadata based focused crawler.
- Location and geographical based focused crawler.
- Incremental and revisit policy based focused crawler.

2. Literature Review

Searching, acquiring and indexing of the Web by search engines can be achieved by using Web crawlers. Web Crawlers are the software programs that traverse the Internet gathering links and information about the web pages they encounter [10]. The relevant information is extracted from the web pages that are downloaded by the page downloader. These web pages are then sorted and indexed. Crawling involves performance and dependability issues since the web is huge. The crawler is required to fetch all the relevant data while maintaining the already downloaded data up to date.

Instead of collecting and indexing all the web pages over web, focused Web crawler [11]

knows its crawl boundaries. It selectively looks out for web pages that are relevant to a pre defined set of topics. It finds those links on the web pages that are likely to be most relevant while avoiding the irrelevant pages on the Web. The aim of a focused Web crawler is to collect all the information related to a specific topic of interest on Web [1]. The study [12] proposes and discusses execution plans for processing a text database using a scan. The method selected has a great impact on the precision and execution time. [13] , [14] did an initial evaluation of various topic driven web crawlers and how to improve their performance.

The keyword query based focused crawler leads the crawling process by using metadata. The keyword data set is used for creating fruitful queries so that the results obtained are reported back to the system. An example of keyword query based focused crawler is an Indian project for tourism and health named Sandhan [15]. This project focuses at identifying the language of a webpage using N-gram technique. For training and testing motive regional and health queries are used. Tang et al. [16] proposes a focused crawling method for medical information relevance and quality of the webpages obtained by the query. They use relevance feedback crawler using query by example. Altingovde et al. [17] built a query engine that allows keywords and queries on the extracted data. A domain specific Web portal is made that can extract information from the backend database. Mukehrjee [18] introduced a WTMS for collecting topic specific data through crawler. Gatial [19] proposed a focused crawling technique based on page relevance. An up-to-date review of various crawling algorithms for focused crawler is given in [20] and [21]. Generally language specific crawlers [22], [23] are used to collect web pages written in their national language. To our information [24] was the first who adopted focused crawler to collect web pages written in a specific language and named it language specific Web crawler. Finding the query interfaces for hidden or Deep Web is an active area of research [25].A SCTWC approach was introduced by Zhang [26] for increasing the crawling performance using fuzzy class memberships. Seyfi[27] proposed a Treasure crawler which uses a T Graph to assign score to each unvisited link. Yajun [28] proposed an improved VSM for improving the performance of focused crawlers. Kumar [29] [30] uses a keyword match based approach for obtaining Indian origin academician information by using various crawling algorithms for fast retrieval of data. Kumar [29] also used the concept of tunnelling initially introduced by [31] to combine it with focused crawling for building digital libraries. A genetic algorithm based focused

crawler is proposed by Goyal [32] for webpage classification to classify webpages as relevant or irrelevant. Yan [33] proposes a focused web crawler based on improved genetic algorithm by redesigning a more accurate fitness function. Farag [34] proposed another focused crawler for societal events.

Table1: Focused web Crawler Studies and Comparative Analysis [3]

Authors	**Category**	**Strengths**	**Limitations**
Chakrabarti et al [1]	Focused crawler based on link, text and url	Compatible with javascript sources	The assumption of linkage locality and sibling locality is not always true, no support for session mechansim
Co et al[35]	Focused crawler based on link, text and url	Ordering the urls to visit important web pages first	The technique is validated only on Stanford website
Pant et al[36]	Focused crawler based on link, text and url	Lexical and linkage analysis for improving the performance	No support for acronyms in cluster labeling technique
Pant et al[37]	Focused crawler based on link, text and url	It uses word both near a hyperlink as well as on entire page	The performance may be improved by using phrases and word synonyms
Diligenti et al[38]	Focused crawler based on context graph, decision tree and DOM	Improved efficiency as compared to traditional crawler	The requirement for reverse links

Shchekotykhin[39]	Focused crawler based on context graph, decision tree and DOM	Uses combined techniques to identify and exploit navigational structures of website	It does not work with ajax applications
Tsay et al[40]	Focused crawler based on context graph, decision tree and DOM	Taxonomy based and keyword based approaches are used to specify user interest	it works only with pure HTML text
Aggarwal et al[41][42]	Learnable focused crawler	Intelligent crawling that learns the characteristics of linkage structure of www	Linkage characteristics used for www are less
Huang and Ye[43][44]	Learnable focused crawler	Initially it uses SVM classifier and once enough webpages are obtained it switches to naive bayes	The initial speed of the crawler is slow
Chung and Clarke[45][46]	Topic specific focused crawler	Uses hash based technique on the url to determine the topic of page and assigning it to a particular crawler	Multiple crawlers may independently encounter the same url
Noh et al[47]	Topic specific focused crawler	It computes the degree of relevance of webpages using Tf-idf entropy and	Subject system used consist of only 400 webpages

		compiled rules	
Qin et al. [48]	Application based focused crawler	Uses meta search enhanced focused crawling and handles the tunnelling problem efficiently	Validation of the proposed technique is domain specific
Chen and Desai [49]	Focused crawler based on context graph, decision tree and DOM	Uses revised context graph to improve recall and it also gets rid of the strict link distance requirement	The proposed study is not validated

We aimed at applying these concepts on the topic of: Indian origin scientist's information retrieval.

Machine Learning based existing crawling work

S.No	Category	Citations
1	Naive Bayes	1,22,23,36,38
2	Support Vector Machine	37,43,44,57
3	Genetic Algorithm	32,33,55,58,59,60
4	Ensemble Methods	53,54,56
5	Decision Tree	38,39,40,49

The results of above citations show that Chakrabarti et al. [1] were the first to use a (Naive Bayesian) classifier to guide a topical crawler. Naïve Bayes works well with text data, is easier to implement and fast to classify but it works on small datasets .It is a weak choice for

guiding a tropical crawler when compared with SVM. Further, the weak performance of Naive Bayes can also be explained by extreme skewness of posterior probabilities generated by it [22,23,36,38]. SVM is suitable for extreme cases as it looks for extremes of datasets. It also works on small datasets but if number of features are greater than number of samples then the method is likely to give poor performance.SVM also does not provide probability estimates. SVM is computationally efficient but it has some disadvantages which degrades its performance for small datasets [37,43,44,57]. Therefore it is proposed to use SVM combined with some other classifier so as to improve the performance and provide optimized results. These are called ensemble learning algorithms. Ensemble classifiers have been shown to outperform individual classifiers on accuracy and robustness as they combine the decisions of various classifiers. Ensemble methods also improve crawling performance as they handle more input variables, noise and outliers [52,53,54,56].

Decision trees can deal with both numerical and categorical data and require very less effort for data preparation. It also handles non linearity but they can be unstable because small variance in the data results in a huge difference in the output [38,39]. Compared with other algorithms, Genetic Algorithms provides higher precision and recall if large number of features are involved [32,33].

2.1 Tools and Technologies

- MATLAB: MATLAB (MATrix LABoratory) is a numerical computing environment having multi-paradigm that supports 4th generation programming language developed by MAthWorks .It is mostly used in applied mathematics and engineering. It has also properties that make it useful in network analysis too. It implements matrix manipulations, data and functions plot, algorithm implementation, user interfaces formation and interaction with programs written in other languages like C, C++, C#, Java, Python and Fortran.

- PYTHON: Python is a widely used programming language for general-purpose programming. It allows the programmer to express concepts in fewer lines of code that might be used in C++ or Java. Its interpreter is available for many operating systems which can be run on a wide category of systems. It supports automatic memory management and multiple programming paradigms.

- WEKA: Weka is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization.

- JAVA: Java is a programming language created by James Gosling from Sun Microsystems (Sun) in 1991. The aim of Java is to write a program once and then run this program on multiple operating systems. The current version of Java is Java 1.8 which is also known as *Java 8*. Java is defined by a specification and consists of a programming language, a compiler, core libraries and a runtime (Java virtual machine) The Java runtime allows software developers to write program code in other languages than the Java programming language which still runs on the Java virtual machine. The *Java platform* is usually associated with the *Java virtual machine* and the *Java core libraries*.

2.2 ML Algorithms

Different machine learning techniques used for classification problems are:

- Decision tree

  A decision tree helps make a decision about the data item. We need to start the process from the root node and keep on answering a particular question at each node and take the branch that match up to the particular answer. This way we can travel across from the root node then to a leaf and then form conclusions in context to the data item.

- Naive Bayes

  It is a classification technique which depends on the Bayes' Theorem. It is based on the assumption that has independence amongst the Predictors. It is a Naive Bayes Classifier which assumes that a particular feature in a class is not directly related to any other feature. Naive Bayes model isn't difficult to build and is really useful for huge datasets. Naive Bayes have outperformed all the highly sophisticated classification methods.

- Support vector machine

  Support Vector Machine is a supervised machine learning method. It is used in solving both regression and the classification problems. It takes the input, does the manipulation of the input and then provides the output.

- K-nearest neighbour

  It can be used for both classification and regression problems. K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases by a majority vote of its k neighbors. The case being assigned to the class is most common amongst its K nearest neighbors measured by a distance function.

- Linear regression

  Linear regression is an approach for modeling the relationship that lies in between a dependent variable 'y' and another or more independent variables that are denoted as 'x' and expressed in a linear form. The word linear shows that the dependent variable is directly proportional to the independent variables.

- Ensemble methods

  Ensemble methods are the meta-algorithms that combine several machine learning algorithms and techniques into one predictive model in order to decrease the variance.

2.3 List of Open Source Crawlers

Name of Crawler	Language Used
Nutch	Java
Crawler4j	Java
Mercator	Java
Larbin	C++
Heritrix	Java
Scrappy	Python

2.4 Performance Metrics for Focused Web Crawler

Harvest Ratio: Brin and Page[2] mentioned the importance of precision and recall. Due to the huge size of Web, it is near to impossible to measure recall for a focused crawler.[1] The measure of the rate at which relevant web pages are acquired and irrelevant web pages are filtered off from the crawling process. This is called **harvest ratio.**

3. Justification for Research

3.1 Motivation

A Web crawler is the main part of any search engine that seeks and acquires the data that can be indexed by a search engine. When we started studying the topic of Web crawler, absence of a complete systematic literature review was a motivating factor. Our study detects various limitations and strategies for crawling the Web. Our study tries to explore various technologies used for Web crawling and will display their comparative analysis.

The area of a focused crawler is evolving at a rapid speed.

Following are some active areas of research [3].

- **Crawler as a service** - Companies that require same data can work together for crawling and later on access the required data together. It would reduce the overall network traffic and would save the resources that are wasted if a single website is crawled repeatedly by different crawlers.
- **Lack of crawling standards** - Website owners often find the crawling activities dubious because of security reasons and bandwidth consumption. A universal standard must be developed to determine which resources of the server are accessible to the crawler. Rules can be designed to enforce the web crawlers to follow robots.txt.
- **Lack of sentimental search engine** - Another area of active research is sentimental search engine that can be used for product reviews, social media applications and marketing etc. prevailing techniques for sentiment analysis use text analysis and NLP to retrieve subjective information of the webpage. But different factors like context,

cultural and linguistic refinements to extract sentiments from the web page can be considered.

- **Crawling multimedia information** – Until now, no existing crawler downloads and processes videos, images and audio files. Crawling and indexing of multimedia data can be an active area of research.
- **Understanding website structure and seed url** - A good website structure helps in faster crawling. Henceforth understanding website structure and best seed url can lead to crawler success.
- **Social network crawlers** - Most of the social media allows very limited access to crawlers. Infact linkedin has publically stated crawling to be illegal. As social media data can be used for decision making, it becomes very crucial to design a plan for crawling it.

As stated above, all areas are open for research.We have identified area of focused crawling based on ML for retrieving information of Indian origin Scientists.

3.2 Research Gaps

- Crawling is a never ending process because of huge size of Web. Some sort of stopping criteria is required to stop the crawling process [29].
- There is a need to develop ML based focused crawler for fast and accurate information retrieval.
- A procedure is required to ensure that the crawler does not move out of the domain specified. For example, any website may have many outlinks of YouTube or Facebook. Such links may lead the crawler to some other domain resulting in wastage of resources and time [29].
- The crawler will crawl all the web pages and fetch the data endlessly if no depth is specified. To stop the crawler after following a pre defined set of URLs, Some depth control criteria need to be designed [29].
- There is a need to improve cluster labelling technique to include labels which should maintain the case atleast for what appear to be acronyms [3].

- A technique is required to evaluate the use of phrases instead of words that are extracted from the link context. Use of synonyms can be helpful for extending the context [3].

4. Problem Statement

To design and develop machine learning based focused crawler to retrieve information of Indian Origin Scientists.

4.1 Objectives

The objectives of the proposed work are as follows:

1. To study existing focused crawler strategies and their relative comparison.
2. To prepare crawling constraints such as seed urls, depth of crawling, domain relevant keyword database.
3. To design and develop a focused crawling technique to retrieve information of Indian origin scientists.
4. To verify and validate the proposed technique with available open source subject systems.

4.2 Methodology

The methodology involves the processes that have to be followed to carry out the objectives of the research. It is intended that objectives can be achieved by implementing following phases of research. To achieve the defined objectives, methodology is divided into four phases. These steps are briefly defined as below:

- The first phase involves fulfilment of Objective 1 which includes the study of various existing focused crawler strategies and their relative comparison. The work of each researcher has been thoroughly studied to develop an algorithm to optimize performance, thereby making the crawling procedure much faster and efficient.

- The second phase involves the preparation of crawling constraints such as seed urls through a tool called SEOQUAKE (an addon of Mozilla Firefox), domain relevant keyword database has to be prepared manually which will consist of names of Indian Institutes, second names of Indians, designations etc so that the system can be trained using these keywords. Weight can be assigned to each keyword depending upon its importance.

- The third phase comprises of design and development of a focused crawling technique or algorithm for information retrieval using Java or Python.

- The fourth phase includes the verification and testing of our technique to optimize performance. For testing, designed crawler is executed for different sets of values and different depths. The data obtained is recorded to study the efficiency and constraints of the code. The data fetched is observed to see at what depth relevant data could be expected. Proper indexing method can be used for fast information retrieval. For this, we can use WEKA tool.

4.3Work Plan

Task	Sep,2019-Nov,2019	Dec,2019-Feb,2020	Mar,2020-Aug,2020	Sep,2020-Feb,2021	Mar,2021-Aug,2021
To study existing focused crawler strategies and their relative comparison.					
To prepare crawling constraints such as seed urls, depth of crawling, domain relevant keyword database Research paper/writing **First progress in Feb 2020**					
To design and develop a focused crawling technique to retrieve information of Indian origin scientists Research paper/writing **Second progress in Sep 2020**					
To verify and validate the proposed technique with available open source subject systems. **Research paper/writing** **Third progress in Mar 2021**					

Communicate the research paper and Thesis writing **Thesis submission in Aug 2021**					

5. Expected Outcomes

The desired deliverables of the work will be

1. A database of the Indian scientists working outside India.
2. Machine learning based focused crawler that will gather and index the desired information.
3. User Interface for extracting information of interest by the users.

- Contributions for the research community

  This search interface of scientists will give Indian students a platform for potential collaborations and interactions. It can be also be used to show case and highlight the achievements of Indian scientists working abroad.

- Potential new technical implications etc.

  The focused crawler can be further scaled to optimize performance, which will make the crawling process much faster and efficient. A proper revisit policy should be devised to continuously update the collected information.

6. References

1. Chakrabarti, S., Van den Berg, M. and Dom, B., 1999. Focused crawling: a new approach to topic-specific Web resource discovery. *Computer networks*, *31*(11-16), pp.1623-1640.
2. Brin, S. and Page, L., 2012. Reprint of: The anatomy of a large-scale hypertextual web search engine. *Computer networks*, *56*(18), pp.3825-3833.
3. Kumar, M., Bhatia, R. and Rattan, D., 2017. A survey of Web crawlers for information retrieval. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *7*(6), p.e1218.
4. Yu, H.L., Bingwu, L. and Fang, Y., 2010, July. Similarity computation of web pages of focused crawler. In *2010 International Forum on Information Technology and Applications* (Vol. 2, pp. 70-72). IEEE.
5. Gravano, L., Ipeirotis, P.G. and Sahami, M., 2003. QProber: A system for automatic classification of hidden-web databases. *ACM Transactions on Information Systems (TOIS)*, *21*(1), pp.1-41.
6. Hammer, J. and Fiedler, J., 2000. Using mobile crawlers to search the web efficiently. International Journal of Computer and Information Science, 1(1), pp.36-58.
7. Kumar, M. and Bhatia, R., 2016, March. Design of a mobile Web crawler for hidden Web. In 2016 3rd International Conference on Recent Advances in Information Technology (RAIT) (pp. 186-190). IEEE.
8. Badawi, M., Mohamed, A., Hussein, A. and Gheith, M., 2013. Maintaining the search engine freshness using mobile agent. Egyptian Informatics Journal, 14(1), pp.27-36.
9. Naghavi, M. and Sharifi, M., 2012. A proposed architecture for continuous web monitoring through online crawling of blogs. *arXiv preprint arXiv:1202.1837*.
10. Pant, G., Srinivasan, P. and Menczer, F., 2004. Crawling the web. In Web Dynamics (pp. 153-177). Springer, Berlin, Heidelberg.
11. Shokouhi, M., Chubak, P. and Raeesy, Z., 2005, April. Enhancing focused crawling with genetic algorithms. In Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on (Vol. 2, pp. 503-508). IEEE.
12. Ipeirotis, P.G., Agichtein, E., Jain, P. and Gravano, L., 2007. Towards a query optimizer for text-centric tasks. ACM Transactions on Database *Systems (TODS)*, *32*(4), p.21.

13. Avraam, I. and Anagnostopoulos, I., 2011, September. A comparison over focused web crawling strategies. In 2011 15th Panhellenic Conference on Informatics (pp. 245-249). IEEE.
14. Batsakis, S., Petrakis, E.G. and Milios, E., 2009. Improving the performance of focused web crawlers. *Data & Knowledge Engineering*, *68*(10), pp.1001-1013.
15. Priyatam, P.N., Vaddepally, S.R. and Varma, V., 2012, November. Domain specific search in indian languages. In *Proceedings of the first workshop on Information and knowledge management for developing region* (pp. 23-30). ACM.
16. Tang, T.T., Hawking, D., Craswell, N. and Griffiths, K., 2005, October. Focused crawling for both topical relevance and quality of medical information. In *Proceedings of the 14th ACM international conference on Information and knowledge management* (pp. 147-154). ACM.
17. Altingovde, I.S. and Ulusoy, O., 2004. Exploiting interclass rules for focused crawling. *IEEE Intelligent Systems*, *19*(6), pp.66-73.
18. Mukherjea, S., 2000. WTMS: a system for collecting and analyzing topic-specific Web information. *Computer Networks*, *33*(1-6), pp.457-471.
19. Gatial, E., Balogh, Z., Laclavik, M., Ciglan, M. and Hluchy, L., 2005. Focused web crawling mechanism based on page relevance. *Proceedings of ITAT*, pp.41-46.
20. Menczer, F., Pant, G. and Srinivasan, P., 2004. Topical web crawlers: Evaluating adaptive algorithms. *ACM Transactions on Internet Technology (TOIT)*, *4*(4), pp.378-419.
21. Avraam, I. and Anagnostopoulos, I., 2011, September. A comparison over focused web crawling strategies. In 2011 15th Panhellenic Conference on Informatics (pp. 245-249). IEEE.
22. Tadapak, P., Suebchua, T. and Rungsawang, A., 2010, September. A machine learning based language specific web site crawler. In Network-Based Information Systems (NBiS), 2010 13th International Conference on (pp. 155-161). IEEE.
23. Srisukha, E., Jinarat, S., Haruechaiyasak, C. and Rungsawang, A., 2008, May. Naive bayes based language-specific web crawling. In Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2008. ECTI-CON 2008. 5th International Conference on (Vol. 1, pp. 113-116). IEEE.
24. Tamura, T., Somboonviwat, K. and Kitsuregawa, M., 2007. A method for language-specific Web crawling and its evaluation. *Systems and Computers in Japan*, *38*(2), pp.10-20.

25. Zhao, F., Zhou, J., Nie, C., Huang, H. and Jin, H., 2016. SmartCrawler: a two-stage crawler for efficiently harvesting deep-web interfaces. *IEEE transactions on services computing*, *9*(4), pp.608-620.
26. Zhang, H. and Lu, J., 2010. SCTWC: An online semi-supervised clustering approach to topical web crawlers. Applied Soft Computing, 10(2), pp.490-495.
27. Seyfi, A., Patel, A. and Júnior, J.C., 2016. Empirical evaluation of the link and content-based focused Treasure-Crawler. Computer Standards & Interfaces, 44, pp.54-62.
28. Du, Y., Liu, W., Lv, X. and Peng, G., 2015. An improved focused crawler based on semantic similarity vector space model. *Applied Soft Computing*, *36*, pp.392-407.
29. Kumar, M., Bhatia, R., Ohri, A. and Kohli, A., 2016, April. Design of focused crawler for information retrieval of Indian origin Academicians. In *2016 International Conference on Advances in Computing, Communication, & Automation (ICACCA)(Spring)* (pp. 1-6). IEEE.
30. Kumar, M., Bindal, A., Gautam, R. and Bhatia, R., 2018. Keyword query based focused Web crawler. Procedia Computer Science, 125, pp.584-590.
31. Bergmark, D., Lagoze, C. and Sbityakov, A., 2002, September. Focused crawls, tunneling, and digital libraries. In International Conference on Theory and Practice of Digital Libraries (pp. 91-106). Springer, Berlin, Heidelberg.
32. Goyal, N., Bhatia, R. and Kumar, M., 2016. A genetic algorithm based focused Web crawler for automatic webpage classification.
33. Yan, W. and Pan, L., 2018, March. Designing focused crawler based on improved genetic algorithm. In 2018 Tenth International Conference on Advanced Computational Intelligence (ICACI) (pp. 319-323). IEEE.
34. Farag, M.M., Lee, S. and Fox, E.A., 2018. Focused crawler for events. *International Journal on Digital Libraries*, *19*(1), pp.3-19.
35. Cho, J., Garcia-Molina, H. and Page, L., 1998. Efficient crawling through URL ordering. Computer Networks and ISDN Systems, 30(1-7), pp.161-172.
36. Pant, G., Tsioutsiouliklis, K., Johnson, J. and Giles, C.L., 2004, June. Panorama: extending digital libraries with topical crawlers. In Proceedings of the 2004 Joint ACM/IEEE Conference on Digital Libraries, 2004. (pp. 142-150). IEEE.
37. Pant, G. and Srinivasan, P., 2005. Link contexts in classifier-guided topical crawlers. IEEE Transactions on knowledge and data engineering, 18(1), pp.107-122.

38. Diligenti, M., Coetzee, F., Lawrence, S., Giles, C.L. and Gori, M., 2000, September. Focused Crawling Using Context Graphs. In VLDB (pp. 527-534).
39. Shchekotykhin, K., Jannach, D. and Friedrich, G., 2010. xCrawl: a high-recall crawling method for Web mining. Knowledge and Information Systems, 25(2), pp.303-326.
40. Tsay, J.J., Shih, C.Y. and Wu, B.L., 2005. Autocrawler: An integrated system for automatic topical crawler. In Fourth Annual ACIS International Conference on Computer and Information Science (ICIS'05) (pp. 462-467). IEEE.
41. Aggarwal, C.C., Al-Garawi, F. and Yu, P.S., 2001. On the design of a learning crawler for topical resource discovery. ACM Transactions on Information Systems (TOIS), 19(3), pp.286-309.
42. Aggarwal, C.C., Al-Garawi, F. and Yu, P.S., 2001, April. Intelligent crawling on the World Wide Web with arbitrary predicates. In Proceedings of the 10th international conference on World Wide Web (pp. 96-105). ACM.
43. Huang, Y. and Ye, Y., 2004, December. wHunter: a focused web crawler–a tool for digital library. In International Conference on Asian Digital Libraries (pp. 519-522). Springer, Berlin, Heidelberg.
44. Kumar, M. and Vig, R., 2012. Learnable focused meta crawling through Web. *Procedia Technology*, *6*, pp.606-611.
45. Chung, C. and Clarke, C.L., 2002, November. Topic-oriented collaborative crawling. In Proceedings of the eleventh international conference on Information and knowledge management (pp. 34-42). ACM.
46. Angkawattanawit, N. and Rungsawang, A., 2002, October. Learnable crawling: An efficient approach to topic-specific web resource discovery. In *2nd international Symposium on communications and Information Technology (ISCIT 2002)*.
47. Noh, S., Choi, Y., Seo, H., Choi, K. and Jung, G., 2004, August. An intelligent topic-specific crawler using degree of relevance. In International Conference on Intelligent Data Engineering and Automated Learning (pp. 491-498). Springer, Berlin, Heidelberg.
48. Qin, J., Zhou, Y. and Chau, M., 2004, June. Building domain-specific web collections for scientific digital libraries: a meta-search enhanced focused crawling method. In Proceedings of the 2004 Joint ACM/IEEE Conference on Digital Libraries, 2004. (pp. 135-141). IEEE.

49. Chen, R., 2008. An enhanced web robot for the CINDI system (Doctoral dissertation, Concordia University).
50. Novak, B., 2004. A survey of focused web crawling algorithms. *Proceedings of SIKDD*, *5558*, pp.55-58.
51. DST Technical Project Completion Report, Dr Rajesh Bhatia, Data Mining And Analysis of Indian origin academicians working in foreign universities for exploring academic collaboration,2018.
52. Pant, G. and Srinivasan, P., 2005. Learning to crawl: Comparing classification schemes. ACM Transactions on Information Systems (TOIS), 23(4), pp.430-462.
53. Saha, S., Murthy, C.A. and Pal, S.K., 2009, February. Rough set based ensemble prediction for topic specific web crawling. In 2009 Seventh International Conference on Advances in Pattern Recognition (pp. 153-156). IEEE.
54. Kim, T.J. and Kim, H.J., 2017. Machine Learning-Based Topical Web Crawler: An Ensemble Approach Incorporating Meta-Features. Journal of Engineering and Applied Sciences, 12(18), pp.4651-4656.
55. Özel, S.A., 2011. A web page classification system based on a genetic algorithm using tagged-terms as features. Expert Systems with Applications, 38(4), pp.3407-3415.
56. Chau, M. and Chen, H., 2008. A machine learning approach to web page filtering using content and structure analysis. Decision Support Systems, 44(2), pp.482-494.
57. Luong, H.P., Gauch, S. and Wang, Q., 2009, February. Ontology-based focused crawling. In 2009 International Conference on Information, Process, and Knowledge Management (pp. 123-128). IEEE.
58. Bai, R., Wang, X. and Liao, J., 2007, June. Combination of rough sets and genetic algorithms for text classification. In International Workshop on Autonomous Intelligent Systems: Multi-Agents and Data Mining (pp. 256-268). Springer, Berlin, Heidelberg.
59. Qi, D. and Sun, B., 2004, November. A genetic k-means approaches for automated web page classification. In Proceedings of the 2004 IEEE International Conference on Information Reuse and Integration, 2004. IRI 2004. (pp. 241-246). IEEE.
60. Liu, H. and Huang, S.T., 2003, August. A genetic semi-supervised fuzzy clustering approach to text classification. In International Conference on Web-Age Information Management (pp. 173-180). Springer, Berlin, Heidelberg.