Vol 9, Issue 5, 2026 **Impact Factor: 8.535** DOI: https://doi.org/10.31426/ijamsr.2026.9.5.9322

IJAMSR

International Journal of Advanced Multidisciplinary Scientific Research (IJAMSR) ISSN:2581-4281

A Comprehensive Study of Web Crawling Algorithms and Their Applications

Pasnur Deeplaxmi Purshottam [1]

[1] Research Scholar, Department of Computer Science Engineering, Mansarovar Global University, Sehore, M.P., India.

Dr. Jaikumar M. Patil [2]

[2] Supervisor, Department of Computer Science Engineering, Mansarovar Global University, Sehore, M.P., India.

ABSTRACT

As the size of the Internet is expanding quickly, it has become vital to improve the search for material quicker and more precise. Without effective search engines, it would be hard to acquire reliable results. To address this challenge, software called "Web Crawler" is employed which employs several sorts of algorithms to accomplish the purpose. These algorithms utilize different sorts of heuristic features to boost efficiency of the crawlers. This research includes a study of web crawling algorithms and their function in enhancing information retrieval. It analyzes many categories of web crawlers. It furthermore examines significant crawling methodologies. The research emphasizes the significance of choosing appropriate crawling techniques to navigate the ever-growing and often fluctuating online landscape. It also discusses how intelligent crawling algorithms may help find important web sites, eliminate wasteful downloads, and enhance the arrangement of online information. The research offers a framework for understanding the strengths and limits of current crawling technologies and their contribution to good search engine indexing.

Keywords: *Web Crawling, Information, Search Engine, Algorithms, Crawler.*

I. INTRODUCTION

The World Wide Web is the most well recognized and crucial repository of information for data mining studies. Consequently, extracting valuable and innovative information and insights from the vast, dynamic, intricately structured, and ever-expanding World Wide Web is a formidable challenge. Online mining is a procedure using data mining techniques to systematically extract and analyze information from online files and services. The primary objective of web mining is to identify valuable information on the World Wide Web and its use patterns. Web mining is often categorized into three types of mining methodologies: Web content mining (WCM), Web structure

IJAMSR

International Journal of Advanced Multidisciplinary Scientific Research (IJAMSR) ISSN:2581-4281

mining (WSM), and Web use mining (WUM). Content mining extracts information from the substance of online documents. Web structure mining extracts structural data from the internet. Web use mining discerns or investigates intriguing usage patterns among vast amounts of data.

The prevalence of online sites is rising daily. We need a search engine to extract information from the World Wide Web repository. Search engines are programs that investigate information from websites. The web crawling algorithm assists the search engine in gathering webpages from the internet. It arises due to the challenge of accessing the vast data kept online. The search engine depends on crawlers to locate relevant sites on the internet.

A web crawler, sometimes known as a spider, is a software application that systematically navigates the World Wide Web in an automated fashion. A crawler, sometimes known as a spider, bot, or agent, is software designed for the purpose of web crawling. The fundamental structure of a web crawler is shown below (Figure 1). Over 13% of a website's traffic is derived from online searches. Currently, the web has billions of web pages, a figure that is too large, with an exponential development rate that poses significant challenges for search engines in managing this vast quantity. The large scale of the online results in limited coverage, with search engine indexing failing to include one-third of the publicly accessible internet. Through the examination of diverse log files from numerous websites, it was determined that web crawlers create the highest volume of online requests, averaging about 50%. Online crawling is not only a programming endeavor; it constitutes a challenge in algorithm and system design due to the vastness of online material. Currently, only Google claims that it has indexed more than 3 billion online pages. The internet has expanded every 9-12 months, and the pace of change is very rapid. Approximately 40% of online sites undergo minor modifications on a weekly basis, however when accounting for changes of one-third or more, the alteration rate is roughly 7% each week.

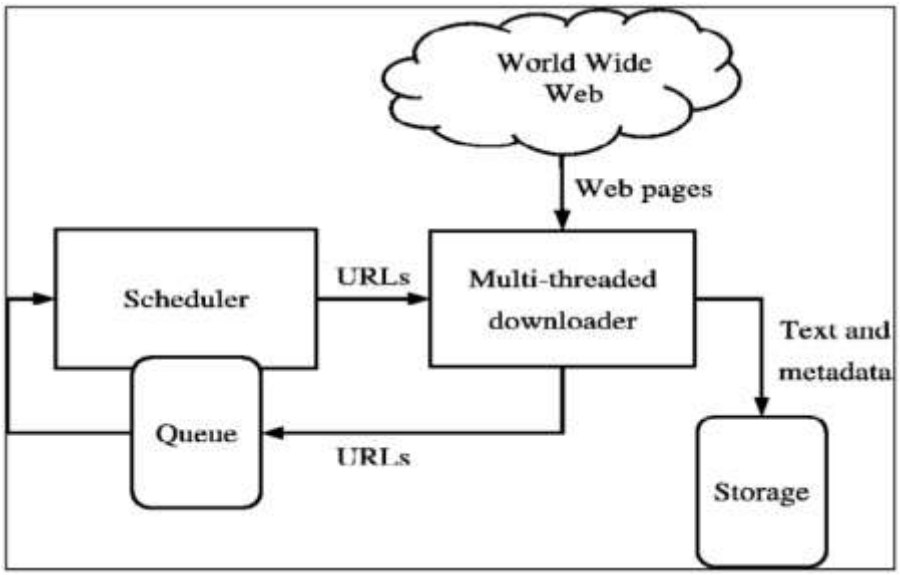


Figure 1: Architecture of Crawler

IJAMSR

International Journal of Advanced Multidisciplinary Scientific Research (IJAMSR) ISSN:2581-4281

Investigators are formulating a novel scheduling strategy for online page downloads that ensures the most significant pages, from the user's perspective, are acquired even if not all pages are downloaded. Since the volume of Internet data expands, prioritizing the download of essential files will become crucial, since downloading everything will be unfeasible.

II. TYPES OF WEB CRAWLER

The way online pages are crawled and future web pages are obtained and accessible determines the sort of web crawler that is available. This is a selection of a few.

Incremental Crawler

One kind of conventional web crawler is an incremental web crawler, which, rather than starting the crawling process from the beginning every time, continuously updates an existing list of downloaded sites. Here you may find a method to check whether a site has been updated since you last downloaded it. Crawling is a continual process that theoretically never stops, and pages might appear in the crawler order several times. With an incremental web crawler, reviewing previously downloaded pages is linked to the initial visit to new pages, so the content of downloaded web pages is always up-to-date. The objective is to accomplish both coverage and updating simultaneously. With an incremental web crawler, you may enhance your data while storing network bandwidth since only useful information is sent to the user.

Form Focused Crawler

The unusually high frequency of online forms is the focus of Form Focused Crawler. Form Crawler learns the features of links and sites that lead to pages with searchable forms, applies suitable stopping criteria, and restricts search to a specified subject so that it avoids crawling through wasteful connections. A web crawler bases its search results on two rankings: site and links. After that, we use a third classifier—the shape classifier—to eliminate forms that aren't necessary.

Focused Crawler

Documents that are narrowly connected to the subject at hand are gathered by a Focused Crawler. As a way of approaching its operation, this crawler is sometimes called Topic Crawler. One kind of web crawler, known as Focused Crawler, has a tendency to transmit related sites. Checks the provided page for any similarities to the specified subject. Economical flexibility in hardware and network resources is one of the focused crawler's features. Downloads, logging, and network traffic are all decreased. By focusing on smaller subsets of the network, Focused Crawler is able to more efficiently seek, acquire, index, and preserve pages. The classifier in this crawler learns to value taxonomy-embedded instances, and the distiller finds the most pressing points on the web right now.

Parallel Crawler

It is becoming more challenging to get the complete or a significant chunk of the web using a single technique as the size of the internet increases. To achieve the maximum download pace, it is common practice for many search engines to operate numerous processes in simultaneously to carry out the aforementioned work. A parallel crawler is the name given to this kind of crawler. Another

IJAMSR

International Journal of Advanced Multidisciplinary Scientific Research (IJAMSR) ISSN:2581-4281

way of putting it is that parallel crawlers are used when several crawlers are often run in parallel. Parallel crawlers use a network of workstations to execute numerous crawling processes, often known as C-procs. Page selection and freshness are the two main factors that the parallel crawlers use. Crawlers running in parallel might be located on the same network or in separate parts of the world. From the perspective of reading and downloading texts in a reasonable amount of time, crawling system parallelisation is crucial.

Distributed Crawler

One approach to distributed computing is Distributed Web Crawler. With the goal of mastering as much online coverage as possible, several crawlers are competing for distribution inside the web crawling technique. Despite its dispersed location, a central server oversees all node communication and synchronisation. Mostly relies on the PageRank algorithm to improve search quality and performance. Distributed web crawlers have the benefit of being adaptable and unaffected by system failures or other occurrences. This makes them ideal for many crawling applications. The creation of the distribution work amongst several computers in a synchronous process is essential for the construction of an effective web crawler. Websites with a lot of content should be spread out throughout the network in a way that allows for synchronous access and makes sense. Additionally, network bandwidth resources are conserved by synchronous dissemination.

III. WEB CRAWLER ALGORITMS

Breadth First Search Algorithm

This algorithm begins at the root URL and explores all adjacent URLs at the same tier. If the objective is accomplished, it signifies success and the inquiry concludes. If it is not, the search continues to the subsequent level, traversing the neighbouring URLs at that tier and so forth until the objective is attained. When all the URLs have been examined, and the goal remains unachieved, it is classified as a failure. Breadth-first search is particularly advantageous in scenarios where the goal is in the upper levels of a deeper tree. Its performance will be suboptimal when the game tree has several branches, particularly in games like chess, and when all paths converge on the same goal with equivalent lengths.

In the illustration of Figure 2, initiate a search for stars from the root URL to gather URLs i, ii, and iii. The process continues by retrieving these three URLs, followed by the subsequent task of downloading i.a, i.b, and i.c as well. Sections ii.a, ii.b, ii.c, and iii.a, iii.b, iii.c are of equivalent hierarchy. Additional URLs i.a.a, i.a.b, i.a.c, as well as ii.a.a, ii.a.b, ii.a.c, and iii.a.a. Versions iii.a.b and iii.a.c have been downloaded.

IJAMSR

International Journal of Advanced Multidisciplinary Scientific Research (IJAMSR) ISSN:2581-4281

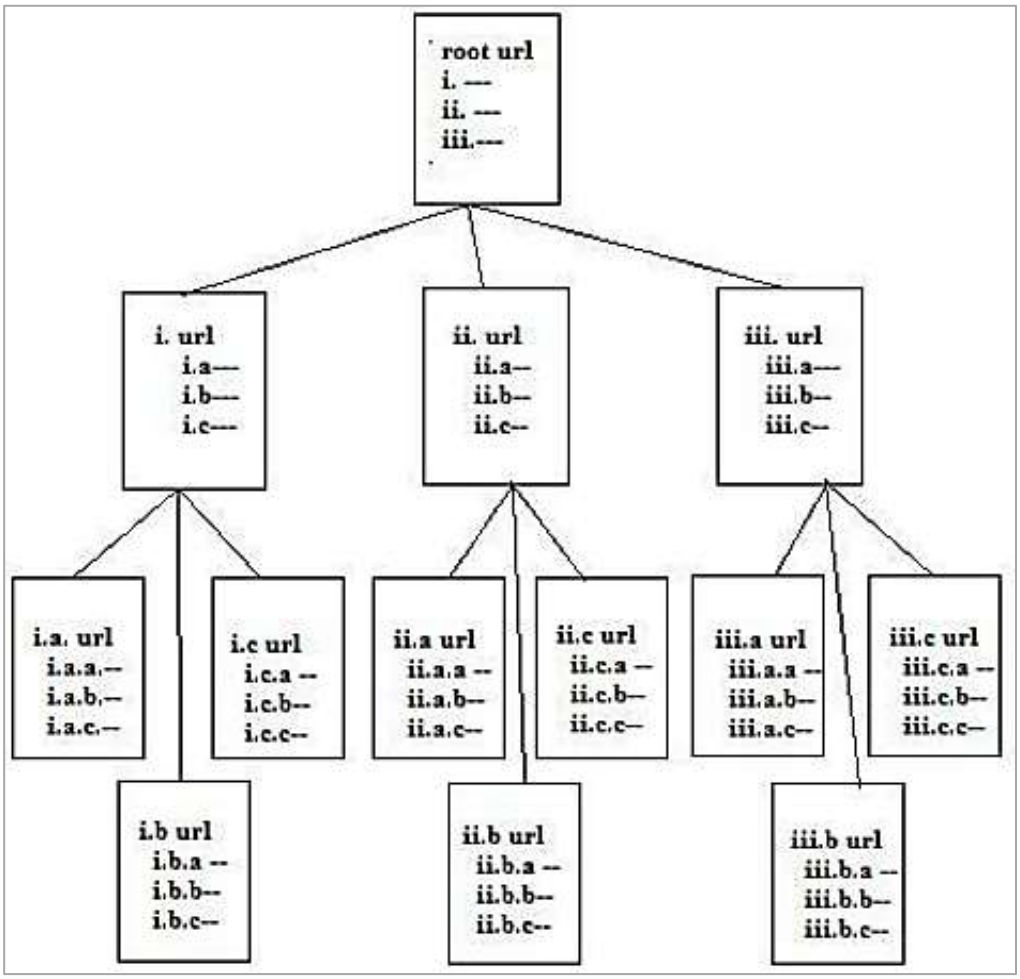


Figure 2: Best First Search and Depth First Search

Depth First Search Algorithm

This robust search begins at the root URL and delves farther into the subordinate URLs. If many children exist, precedence is given to the leftmost kid, and one should go deeply until no more children are present. It reverts to the subsequent unvisited node and proceeds in a like fashion. This approach ensures that every edge is traversed once in a breadth-first manner. This technique is very appropriate for search problems; nonetheless, it may enter an endless loop when the branches are extensive.

In the illustration in Figure 2, the search starts at the root URL and retrieves the first i URLs. The search advances by downloading i.a followed by i.a.a, i.a.b, i.a.c, then i.b and its offspring, and then i.c and its offspring. Subsequent to this phase, the ii URL is retrieved and the procedure advances.

Page Rank Algorithm

The Page Rank algorithm considers the interconnections among online pages. For instance, if page P1 has a hyperlink to page P2, it is likely that P2's content is attractive to the developer of P1. Consequently, the quantity of inbound links to a webpage indicates the level of attention it garners from others. Evidently, the level of interest in a page escalates with the increasing quantity of inbound connections.

$PR(P1) = PR(A1)/L(A1) + ... + PR(An)/L(An)$

IJAMSR

International Journal of Advanced Multidisciplinary Scientific Research (IJAMSR) ISSN:2581-4281

Page Rank (PR)(P1) may be determined by collecting all the inbound links to and outbound links from a given page (P1). If page A1 has a link to page P1, then page L(A1) will display the number of outbound links to page P1. All the pages that link to Main page P (A2, A3, etc.) undergo the identical procedure, and the sum of their values determines their rank. Furthermore, it is expected that a web site with connections from prominent pages would have a high rank. Based on the weighted total of the input links, the site Rank of a web site may be determined.

Path-Ascending Crawling Algorithm

From the root of a URL all the way to its final file, the path ascending crawling algorithm would explore every possible route. Because of the way the crawler works, it is able to extract more data from the website. Following the steps outlined above, a crawler may reach all paths inside a given URL. It will try to crawl all three of these URLs when given a seed URL like http://rashmi.org/rash/profit.html.

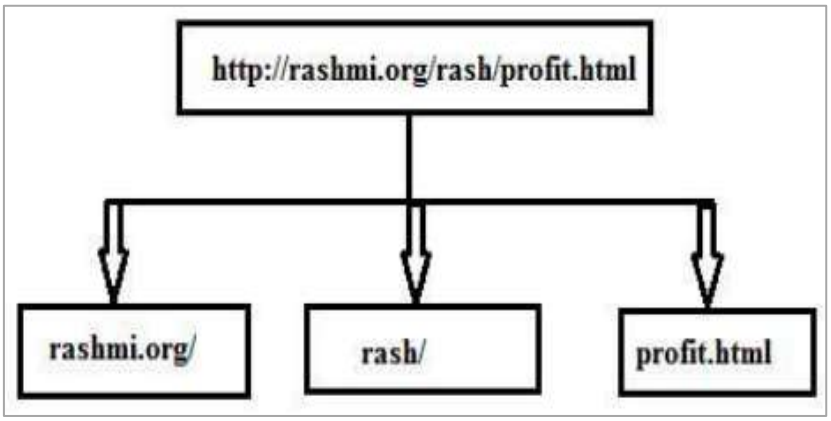


Figure 3: Path - Ascending Crawler

When compared to traditional crawling methods, path-ascending crawlers excel in discovering isolated resources—those without any inbound links—which would otherwise go unnoticed.

Focused Crawling Algorithm

If a site is very similar to a specific query, then it is likely that crawlers will find it significant. This method allows us to program web crawlers to only download related sites; as a result, they are known as focused crawlers or topical crawlers. Focused web crawling has one major flaw: before downloading a site, we'd want to have a sense of how well the text matches the query. By analysing the content of previously visited sites, this crawler may deduce the similarity between the driving query and unvisited pages. To determine how similar an unvisited page is to a visited one, we utilise attributes like URL and anchor text that are accessible without downloading the page specific. Crawling with a specific goal in mind sometimes uses the "seed URLs" provided by a generic web search engine. It is possible to utilize this crawler type to create file type-specific search engines.

Online Page Importance Calculation Algorithm

With this approach, known as Online Page Importance Computation (OPIC), the starting point for all pages is a fixed cash value of 1/n, and this value is then dispersed evenly across all output links. Although it only takes one step, this is quite similar to Page Rank.

IJAMSR

International Journal of Advanced Multidisciplinary Scientific Research (IJAMSR) ISSN:2581-4281

If http://rashmi.org has ―m‖ no. of pages in it, then each page obtains 1/m cash.

At each level, the crawler will download web sites with greater cashes and spread the cash among the pages it links to. In terms of crawling tactics, OPIC is in a league of its own. The method's drawback is that it increases crawling time by downloading each page many times.

Naïve Bayes Classification Algorithm

Probabilistic learning and classification are the basis of the Naïve Bayes method. It takes for granted that each characteristic exists apart from the others. Despite the fact that this algorithm's basic premise is not very relevant in real-world scenarios, it was efficient compared to many other ways.

Dunren Che, Abdullah Althagafi, and Mejdl S. Safran suggested a Naïve Bayes-based crawler that could efficiently collect many relevant sites. Four factors related to relevancy have been considered: the words in the URL, the text of the anchor, the text of the parent sites, and the text surrounding the URL. To further enhance prediction accuracy, additional relevant terms are found and added using WordNet, a free online lexical database.

Sematic Web Crawler Algorithm

Search engines are able to swiftly respond to user query requests because of a process known as crawling, which indexes websites. A specialized query processor may remove stop words and stem the question terms to improve the query. Incorporating semantic approaches into search engines is expected to provide more relevant and specific results.

IV. CONCLUSION

Within the scope of the work that is now being reviewed, an investigation of web crawlers and various web crawling strategies has been carried out. By beginning at the root URL and working its way down via the child URLs, the Depth first search algorithm is a highly popular and widely used searching algorithm. First, it moves to the kid on the left in the event that there is more than one child present, and then it continues to go deep until there are no more children available. The technique of backtracking to the next unvisited node is used in this instance, and processes are compensated in a way that is analogous. The use of these algorithms guarantees that each and every edge, or URL, gets visited once over the expanse of the network. When it comes to searching, it is highly effective. Information on query processing on the web throughout the search process is also provided by the research.

REFERENCES

1. M. Khder, “Web scraping or web crawling: State of art, techniques, approaches and application,” International Journal of Advances in Soft Computing and its Applications, vol. 13, no. 3, pp. 145–168, 2021.
2. K. Velkumar and P. Thendral, “Web crawler and web crawler algorithms: A perspective,” International Journal of Engineering and Advanced Technology, vol. 9, no. 5, pp. 203–205, 2020.

IJAMSR

International Journal of Advanced Multidisciplinary Scientific Research (IJAMSR) ISSN:2581-4281

3. Shrivastava, "Enhance the security and improve the performance of web crawlers using web crawling algorithms," International Journal of Recent Technology and Engineering, vol. 8, no. 4, pp. 11642–11651, 2019.
4. Kumar and R. K. Singh, "Web mining overview, techniques, tools and applications: A survey," International Research Journal of Engineering and Technology, vol. 3, no. 12, pp. 1543–1547, Dec. 2016.
5. Nigam, "Web crawling algorithms," International Journal of Computer Science and Artificial Intelligence, vol. 4, no. 3, pp. 63–67, Sep. 2014.
6. R. Kumar, A. Jain, and C. Agrawal, "Survey of web crawling algorithms," SSRN Electronic Journal, vol. 3, no. 3, pp. 1–7, 2014.
7. M. A. Kausar, V. Dhaka, and S. Singh, "Web crawler: A review," International Journal of Computer Applications, vol. 63, no. 2, pp. 31–36, 2013.
8. J. Akbari Torkestani, "An adaptive focused web crawling algorithm based on learning automata," Applied Intelligence, vol. 37, no. 4, pp. 586–601, 2012.
9. S. M. Pavalam, S. V. K. Raja, F. K. Akorli, and M. Jawahar, "A survey of web crawler algorithms," International Journal of Computer Science, vol. 8, no. 6, no. 1, Nov. 2011.
10. Castillo, "Effective web crawling," ACM SIGIR Forum, vol. 39, no. 1, pp. 55–56, 2005.
11. F. Menczer, G. Pant, and P. Srinivasan, "Topical web crawlers: Evaluating adaptive algorithms," ACM Transactions on Internet Technology, vol. 4, no. 4, pp. 378–419, 2004.