International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 06 | June - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

RESEARCH ON WEB CRAWLER'S AND ITS PERFORMANCE VIA EXAMINATION OF SINGLE AND MULTI-THREADING BY USING ALGORITHMS

CHIRITHOTI MANEESHA[1], G VENU GOPAL[2]

[1]PG Schoolar, Dept. of Computer Science and Engineering, PBR Visvodaya Institute of Technology & Science Autonomous, Affiliated to JNTUA, Kavali, SPSR Nellore, A.P, India-5242201.
[2]Associate Professor, Dept. of Computer Science and Engineering, PBR Visvodaya Institute of Technology & Science Autonomous, Affiliated to JNTUA, Kavali, SPSR Nellore, A.P, India-5242201.

Abstract - These days, while thinking about how to build the World Wide Web, web crawling is a crucial idea. Developing a reliable crawler system that returns relevant and efficient search results for common queries is an ongoing need. On a daily basis, individuals encounter the issue of search results that include unsuitable or inaccurate responses. Therefore, it is crucial to find better ways to provide users accurate search results in a reasonable amount of time. Web users and analyzers may get less useful and perhaps irrelevant findings since not all sites can be visited in less time. One kind of robot that can access hyperlinks and explore online pages is called a web crawler, often called a spider. The documents that are traversed from a web page are gathered in a web pot. The process begins with the seed URL, which is where the user gets the web content. If there are any new links in the downloaded documents, it will delete them. The crawler verifies whether the user has already downloaded the file when the URL is deleted. A web crawler follows the links on a website to read the documents contained within. Developing a reliable crawler system that returns relevant and efficient search results for common queries is an ongoing need. On a daily basis, individuals encounter the issue of search results that include unsuitable or inaccurate responses. Therefore, it is crucial to find better ways to provide users accurate search results in a reasonable amount of time. We demonstrate an efficient method for developing a crawler that takes aspects into account in this project. Moreover, there are a lot of crawlers that visit the seed URL, read the pages, and then download them to add to search engine indexes. An issue arises when the crawler continues to access out-of-date websites or pages, even if they were downloaded on their prior visit. A lot of time, space, bandwidth, and network resources are wasted because of this. It is possible to retrieve pages using a variety of algorithms. This study proposes a new method for web crawlers that use clustering techniques: single and multithreaded web crawling and indexing algorithm. Therefore, you should try to make a good system with a revised policy for web crawlers in order to reduce the occurrence of these issues. After sorting websites into "frequently," "frequently," and "static" in the first scan, the crawler determines when it needs to crawl that same page again. Results from experiments demonstrate that, compared to current approaches, the suggested algorithm achieves optimum execution time. Building an intelligent crawler that can learn to improve the effective ranking of URLs using a focused crawler is the main focus of this project. Initially, links are crawled from specific Uniform Resource Locators (URLs) using a crawling algorithm. This allows users to perform hierarchical scanning for their respective web links.

Key Words: *Web crawler, URL's, Multi Threaded Crawling, Web Crawling Tree, Algorithms.*

1.INTRODUCTION

Using existing databank organisation outfits and antiquated data control demos is difficult when dealing with big data, which is a massive and complicated cluster of data crowds. A term for extremely large datasets is "big data." Engaging in a massive, diverse, and complex arrangement with the difficulties of loading, analysing, and forecasting in order to promote the approaches or outcomes. Big data analytics is the process of sifting through massive amounts of data in search of hidden patterns and correlations. Businesses and government agencies can benefit greatly from this data since it allows them to learn more about their customers and get an advantage over their competitors. The three defining characteristics of Big Data are its versatility, scalability, and visibility.

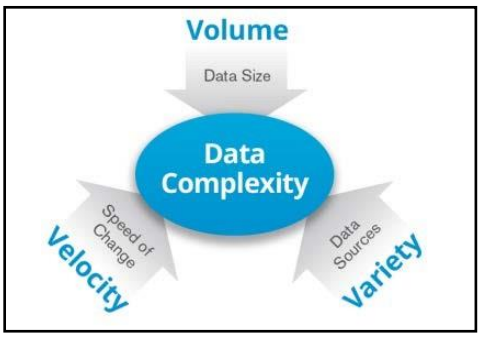

Fig -1: **Three V's of Big Data**

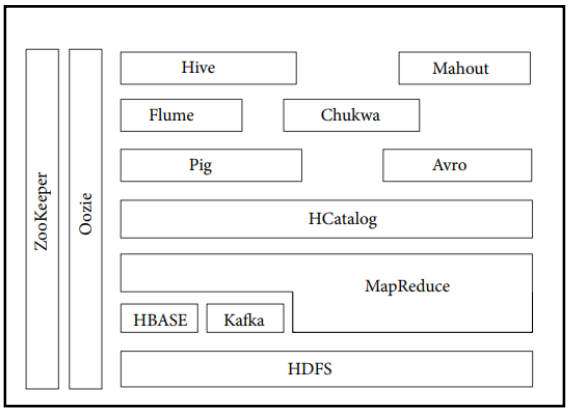

Fig -2: **Architecture of Big Data**

1.1 Web Crawling

One essential and integral framework of the internet is the Search Engine. Users can access limitless data in few seconds without investing much time using search engines. They are a huge source of information and one of the most dominant platforms for business and market. It is a software or script that searches documents, files, images and multimedia contents based on keywords and phrases supplied by the users and returns the results in the form of hyperlinked files containing those keywords or phrases. The returned information is a blend of web pages, graphics, images and some other types of information. Various types of search engines are available now a day with its own quality and abilities. Now search engine is a versatile source of information retrieval and a popular platform for marketing.

Web Crawler is an important component of the search engine that continuously crawls the web to get the pages available in World Wide Web. It is a simple automated program where internet pages can be crawled to retrieve information from web data. It includes web spider, web robot, crawler and automatic indexer. A web document contains graph structure that can be connected through hyperlinks. Figure 3 represents graph structure of hyperlinked web documents.

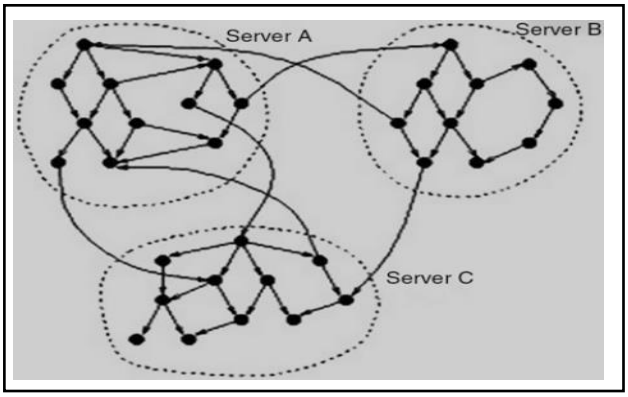

Fig-3: Graph Structure of Hyperlinked Web Documents

The crawl manager starts crawling gradually from specified set of URLto fetch and scan new URL in an endless cycle. A newly identified URL from next cycle process has extracted in and out link of respective web pages. These visited pages are stored in the buffer for further process. An out link web pages are stored to visit the frontier list that can be classified using ontology editing tools. An indexed method is utilized to improve the efficiency of web search. Both HTML and XML web page contents are parsed using parser method. An inverted matrix is constructed using an interpreted data system. It includes a number of occurrence specific words and location of text in particular document. Keywords are constructed for search from inverted index to enhance information retrieval by utilizing ontology classification. Generally, search engines are software methods to retrieve information from the internet. A web crawler has capability to visit web pages on internet to classify and index both current and new pages. The quality of web crawler affects quality of information search directly. Web documents are combined to connect different resources by multiple hypertexts.

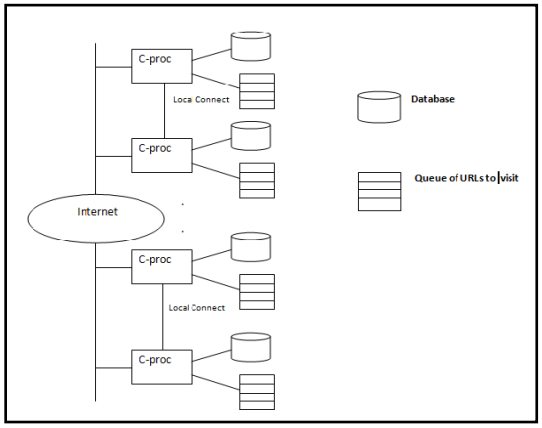

Fig-4: Parallel Crawler

1.2 Objectives

- To find out, improving web crawler speed, we offer a clustering-based method for web crawling and indexing that may operate on single or multiple threads.
- To find out how many sites and links the suggested algorithm crawled in total.
- To locate URLs that the suggested crawling technique has visited.
- To assess how long it takes for a web crawling algorithm to run and how long it takes to gather data.

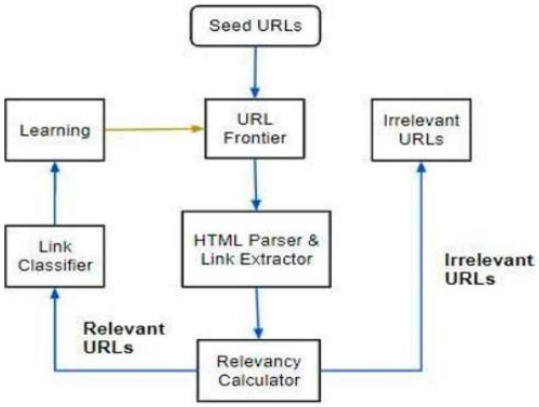

Fig-5: Architecture of the learning crawler

2.PROPOSED WORK

The problem of web crawling has been identified on the basis of past literatures i.e. it's impossible to perform and analyze multiple numbers of web pages at same period of time, because information located in World Wide Web becomes very terrific. The drawback of existing web crawlers is that it does not have the capability to visit and parse every available web page because network bandwidth is very low. One more drawback is that crawler tries to keep multiple data in disk that causes less accuracy, scalability and flexibility.

Some other dilemmas are that web is gigantic and does not follow any specific structure, its size cannot be measured, contents are added, modified and deleted very rapidly, duplicate contents are present, multimedia contents are difficult to crawl and only 15-20% data is available at surface level so it's very difficult to crawl deep and hidden web by crawlers.

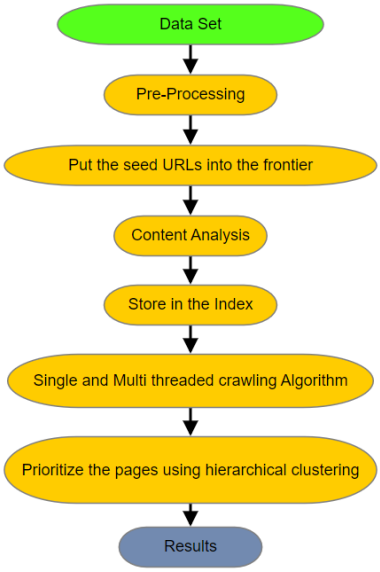

Fig-6: Work Flow of Proposed Methodology by Visirule-365

The above figure 6 depicts the complete workflow of our proposed methodology based on web crawling system. The Dataset DMOZ URL has been collected and undergoes pre-processing phase where the raw data is cleaned and removes irrelevant information then inspected to make it ready for further processing of different web pages. By using DMOZ URL dataset, the frontier among data has initialized by means of seed URLs.

- The suggested crawler's flow is shown in the architecture by various coloured arrows.
- When creating a new query, you should follow the pink, blue, and green arrows in the corresponding way.
- The three arrows—pink, black, and green—are followed in the corresponding way for a registered inquiry.
- The modules in the background that are executing in parallel are shown by red arrows.

This page contents has scrutinised and stored in the index of web crawling system. The pages should be displayed either in HTML or text format. Based upon the requirement type, the specific page contents are accessed by utilizing single and multi-threaded web crawling and indexing algorithm using clustering technique. After that the results of web page contents are arranged by utilizing hierarchical clustering and the harvest ratio, harvest time, and execution time are computed. The obtained results achieve better performance of the proposed algorithm.

2.1 Proposed Algorithm

The process of maintaining the set of unvisited URL in crawler is known as frontier. This list of web page has starts with seed URL which might be provided by user. Every loop of web page implicates harvesting the next to crawl from frontier and fetching the page parallel to specific information through HTTP. To extract specific information of URL, the retrieved pages have been parsed accurately and all unvisited URLs should be added in frontier. Beforehand, the URL that has already been included into the frontier might be considered for score so that the benefits of visiting web page parallel to URL could be estimated.

If the specific number of web pages has been crawled, then crawling process might be terminated. If there are no more pages to crawl and frontier is empty, then this state of crawler can be assumed as a dead-end. This crawling process could also be observed as the graph search problem. At nodes, the huge number of web page graph could be perceived and hyperlink as edges. It begins with few seed nodes and follows edges in order to reach nodes in web pages.

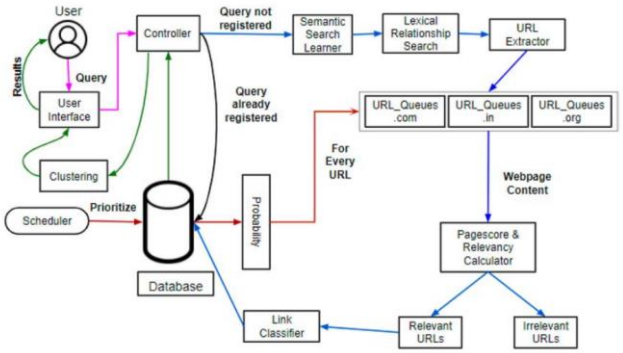

Fig-7: Architecture of Learning-based Focused Web Crawler

3.FRAME WORK FOR SINGLE AND MULTI-THREADED WEB CRAWLING

Data may be retrieved via the internet using the search engine programmes. A web crawler may access any and all web sites on the internet, identify and index old and new ones alike. The information quality that is returned by a web crawler is closely related to its quality. Web crawlers are basic automated programmes that may be used to obtain information from the web by crawling and deploying web pages. The combination of several hypertexts in a web content allows it to link various resources. The inability of a single instance to crawl the whole web is directly proportional to the growing size of the web. In order for search engines to cover the most portion of the web, many simultaneous processes are run. Rather of relying on a single process crawler, it makes use of a multitude of them. It is recommended to save such a website on your local computer rather than viewing it online. The next step is to extract the uniform resource locator and then follow the links that go with it.

Web pages with a thick tree structure that can only be accessed via associated hypertext links are crawled by a publicly indexable web crawler. When it came time to need permission or previous registration, surface web spiders completely ignored both the search forms and the sites.

It disregards a plethora of high-quality information from designated websites. To display web page search results, the concealed contents of each page have been analysed. If it's there, it may be extracted and used to directly populate URLs with information.

Link extraction for individual web pages has begun using a freshly discovered URL from the subsequent cycle. These websites that users have visited will be saved for future use. Frontier stores the outbound links so you can access them on your next visit, and you can use the ontology editing tools to categorise them. To make online search more efficient, an indexed technique is used. You may use the parser technique to read web pages that use either HTML or XML. An interpreted data system is used to build an inverted matrix. It contains the text's position and the number of times it appears in a particular document. Through the use of ontology categorization, it is possible to construct keyword searches using inverted indexes in order to improve data.

3.1 Working of Web Crawling

As a human being in the era of day-to-day's advanced technologies, World Wide Web is an essential part of daily routines. Users using internet are being monitored in some ways or other with the purposes of recommending contents, data mining, web crawling, web scrapping, business and marketing etc. Specifically, web crawling is playing a significant role now days, which is growing with the improvements seen in the WWW and found to be somewhat beneficial in obtaining relevant contents that user desired to get.

There is a great scope for the web crawling since the contents that we tend to access these days are drastically increased, which also posed irrelevant contents for consumers to analyse it. The main components of web crawling process are indicated in the figure 8.

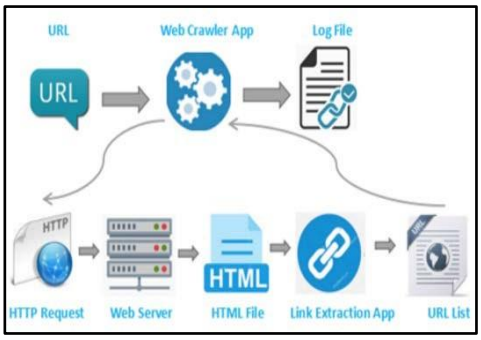

Fig-8: Components of Web Crawling Process

The web robots tend to explore the WWW to find the relevant data as it is needed by the person who wants to explore or search required data for comprehensive analysis. Since the architecture of WWW found to be descriptive in nature, there would be many linked or quoted URLs in the specific URL itself. This architecture of WWW is well utilized by the web robots which enables one to pass through and get to know numerous URLs by accessing the single origin URL itself. Web crawlers are designed to acquire numerous URLs and store it in its intended treasury or repository. After storing URLs, fetching of the identical URLs take place and transferred them for indexing.

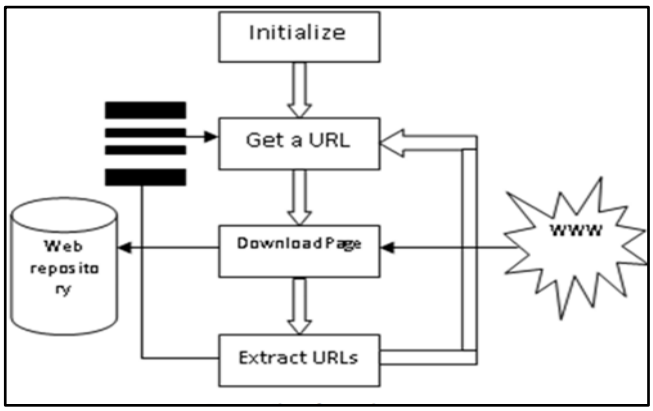

Fig-9: Components of Web Crawling Process

In order to get the desired user-generated material, web crawling methods are often used, as shown in figure 9. Web crawlers often take the form of scripts that allow users to systematically traverse the World Wide Web. You may go to another website by clicking the link. Nodes and hyperlinks traverse the values of a directed graph. Each page that has been viewed before may have a duplicate made using it. The search engine has analysed these visited websites. In order to get content from the World Wide Web, search engines must store information about various kinds of web sites. The three parts that make it up are the online repository, the border, and the page downloader. In this case, the crawler border is the first component. Crawlers use this list of unvisited URLs as a checklist of things to do. There is a list on the website that shows how many seed URLs a user may provide. From the seed URL, the web crawler begins to work. The unvisited web page is added to frontier after retrieving its appropriate URL from the web page. This loop will keep retrieving and extracting URLs until the border is empty or until some other condition is met. The extraction of the border URL is dependent on priority scheduling.

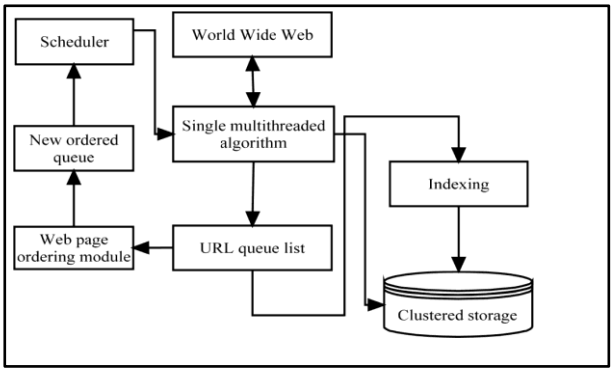

Fig-10: Proposed Working Architecture of Web Crawling Approach

Ordering the web page modules for crawled websites and sorting URL web pages is dependent on link score. The next step is to use an indexing technique to gather all of the connected documents from the web and then to index them. At last, the crawling system estimates a web page's index and stores it in the clustered web storage. In order to avoid parsed URLs, the frontier time control mechanism is internally delayed for a long period. For every new insertion process of parsed URLs, it also controls the waiting time of URLs inside frontier. The time control approach is used to crawl all of the URLs inside the frontier that are waiting for a certain time.

If the URL arrives at the correct moment, this webpage gets removed from the border structure. There should be a maximum initialization time for the waiting time for crawled URLs. Without it, the boundary would be sunk. The proportion of relevant pages downloaded is a good indicator of a web crawler's performance. Performance metrics such as harvest ratio, time, and crawling time are calculated with the aid of the suggested work. We identify and compare the outcomes of our suggested method with those of the current crawling methodology.

3.2 Multi-Threaded Crawling Method

There are various mechanisms that are utilized to make sure that different threads visits multiple host at specific period of time such that every host would not be overloaded by several web page requests in URL. The out links that are not original to the specified host have been dispatched to appropriate user which makes the queue of web pages to be visited. The entire web page in breadth first search has been visited as soon as innovative host has been met. It is possibly estimated with bounds on depth reached or complete number of pages again in a breadth-first approach.

If the queue has no such URL to search, then the process of crawling gets terminated. The process of fetching and extracting links are compared to magnify vertex node in graph search problem in web crawling. An instance for seed URL web pages is given below.

http://www.dmoz.org/Computers/Internet/Cloud_Computing/Service_Providers/
http://www.cloudreviews.com/
http://www.cloudservicemarket.info/services/servicesBrowse.aspx
http://talkincloud.com/tc100
http://atechjourney.com/list-of-free-cloud-storage-services.html
http://compixels.com/2303/list-of-top-free-cloud-based-services
http://www.crn.com/news/cloud.html
http://en.wikipedia.org/wiki

3.3 Hierarchical Clustering

In the proposed algorithm hierarchical clustering technique will be used to link related web pages together. Figure 4.5 depicts the hierarchical clustering of web crawling algorithm. Here, an agglomerative hierarchical clustering in web crawling has been utilized to create a set of nested clusters like tree structure. It is also known as cluster tree. A single cluster in web crawling is located at the bottom of tree.

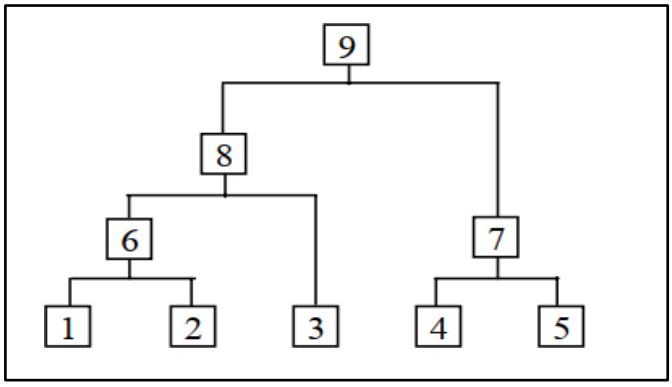

Fig-11: Hierarchical Clustering

Each web page has been covered by an individual root cluster which is situated at the top of crawled process. Next, this clustering process starts constructing the cluster tree from bottom level. At every level, the nearest part of cluster pairs have been estimated till each page has been merged into single clusters. In this diagram, the cluster 5 web pages have been located at the bottom of cluster tree. By combining both clusters 1 and 2, the cluster 6 web pages could be formed such that it is parallel to one another.The web pages of cluster 8 have been formed by combining both clusters 6 and 3. Likewise, the web page of cluster 7 has been formed by merging both clusters 4 and 5, so on. As moves up the cluster tree gradually, there would be only fewer and fewer number of clusters in web page that are crawled. The user could select to view different number of clusters at any tree level, because the entire clustering tree has been stored in clustering process of web page.

The reason for utilizing this hierarchical clustering method is that the cluster tree enables the user to view details of clusters at any level. It creates user analysis and visualization of two web sites very convenient.

4.RESULTS ANALYSIS AND DISCUSSION

The dataset used in web crawling system is DMOZ URL that contains two sources. First is the DMOZ URL which is an open directory project where entries are scrutinised by web page user in URL document. It is responsible for user to precede huge part of directory because it gains both trust and experience. Next, the random URL selector of yahoo directory is predicted by particular user in URL. It is the largest and most wide-ranging part of user-edited web directory in crawling system. The overall community of volunteer editors could be built and preserved by open directory project. It enables user to provide resource to organize internet by itself. Due to the growth of internet, the number of net users is also get increases. A small part of web could be organized for every user. The remaining number of users could be presented back by cutting out unnecessary content and including optimal content.

The movement of open source can be done by DMOZ. It is one of the most popular search engines that contain portals, AOL Search, Google, Lycos, Hotpot, and so on. It has contributed in three following ways-

- The user reports a problem along with sites that are listed.
- The specific category site of a user can be suggested.
- The user offer to create particular category.

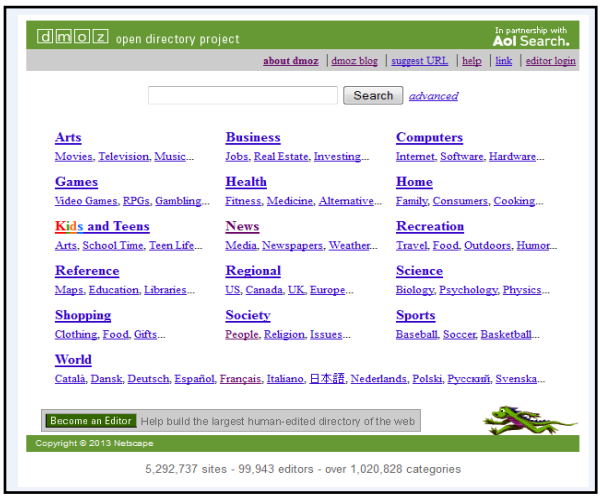

Fig-12: DMOZ URL Open Directory Project

S.No	Cluster Document	Execution Time	Harvest Time
1	Document/<html>	150	95
2	Document/<html>/<body>	150	95
3	Document/<html>/<body>/<h1>	300	95
4	Document/<html>/<body>/<h1>/tech	300	95
5	Document/<html>/<body>/ 	450	95

Table 1: Clustered Document in Page1.Txt

S.No	Clustered Document	Execution Time	Harvest Time
1	Document/<html>	150	95
2	Document/<html>/<body>	150	10
3	Document/<html>/<body>/<h1>	150	10
4	Document/<html>/<body>/<h1>/world	300	95
5	Document/<html>/<body>/ 	450	10
6	Document/<html>/<body>/list	450	10

Table 2: Clustered Document in Page2.Txt

S.No	Clustered Document	Execution Time	Harvest Time
1	Document/<html>	300	95
2	Document/<html>/<body>	150	95
3	Document/<html>/<body>/<h1>	100	10
4	Document/<html>/<body>/<h1>/local	155	10
5	Document/<html>/<body>/ 	400	55
6	Document/<html>/<body>/list	450	55

Table 3: Clustered Document in Page3.Txt

S.No	Clustered Document	Execution Time	Harvest Time
1	Document/<html>	150	10
2	Document/<html>/<body>	300	15
3	Document/<html>/<body>/list	450	55

Table 4: Clustered Document in Page4.Txt

S.No	Clustered Document	Execution Time	Harvest Time
1	Document/<html>	150	95
2	Document/<html>/<body>	150	10
3	Document/<html>/<body>/<h1>	300	10
4	Document/<html>/<body>/<h1>/tech	300	10
5	Document/<html>/<body>/ 	450	55

Table 5: Clustered Document in Page5.Txt

S.No	Clustered Document	Execution Time	Harvest Time
1	Document/<html>	150	95
2	Document/<html>/<body>	150	95
3	Document/<html>/<body>/list	300	10

Table 6: Clustered Document in Page6.Txt

The clustering of HTML documents in the proposed web crawling system contains three main algorithms such as Text MDL, Single and Multi- Threaded Web Crawling and Hierarchical Clustering by using Jaccard and dice coefficient. Generally, clustering plays a very important role for analysing and determining the unsupervised learning problem through the set of unlabelled structure of data in the HTML document. It is the process of combining objects into specific groups that are similar to some process in web crawling system. The purpose of clustering of HTML documents is to replace objects into groups or clusters that have been determined by the particular HTML link of a crawled web page. In some case, clustering of HTML document of a particular path may possess either similar or dissimilar page of cluster that depends on number of page that are crawled in the clustering process. It provides more benefit to explore information into specific web page document in HTML. Clustering is defined as the set or group of abstract objects that are divided and grouped into several similar URL objects in the system. In order to recognize similar group of clusters, a pre-processing data has been estimated as a valuable resource for various HTML clustering documents in the web crawling system. It is necessary to recognize web page or groups that are necessary to develop the supervised and unsupervised models of a clustering HTML web page documents. The main advantage of clustering HTML document through classification is that the changes that occurred are adapted to differentiate various features of clusters in the web crawling. It is defined as a process of collecting and determining variouslevels of documents through which set of documents are stored into the bins that can be performed as a function of particular clustered web page in URL. This crawled web page has grouped and separated into subset of clusters. A clustering of HTML documents is divided into two main types as unsupervised linear clustering and non-linear clustering algorithms. An unsupervised linear clustering algorithm allows data to be linear for particular clustered algorithm that enables to find hidden patterns or grouping data. Likewise, an unsupervised non-linear clustering algorithm allows the clustered algorithm to be non-linear. A clustering can be done by using text MDL, Min Hash Jaccard Coefficient, and Min Hash Dice Coefficient in order to identify and calculate probabilistic of a proposed hierarchical clustering of URL in web crawling. The Clustering by Min Hash Jaccard Coefficient and Min-Hash Dice coefficient takes web documents and essential paths in each of the web documents as its input. Each of the web documents is considered as a cluster and the signature values are computed for each cluster. The documents with the same signature values are then clustered together. The difference between these two clustering methods is that the Min Hash Dice coefficient uses Dice coefficient while the Jaccard coefficient method uses Jaccard coefficient. The paths maximum jaccard coefficient and maximum dice coefficient are taken and the values of their crawled web page are computed. The web crawling method for clustering takes web documents and its essential paths as input and determines the values of harvest and execution time. Once the clustering is performed value for each web page in hierarchical clustering can be computed. The method which shows least value of URL page is chosen as the best clustering model.

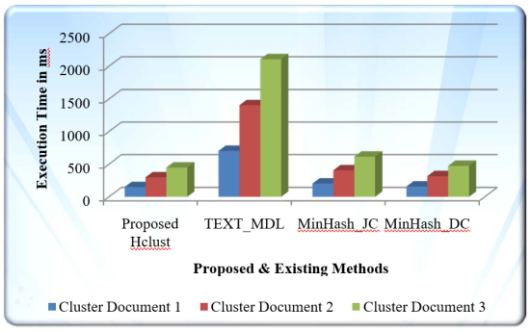

Fig-13: Performance Analysis of Execution Time

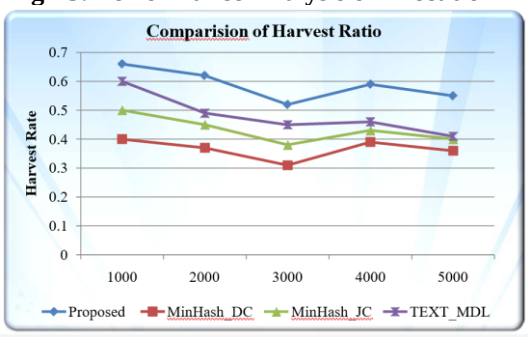

Fig-14: Performance Analysis of Harvest Ratio

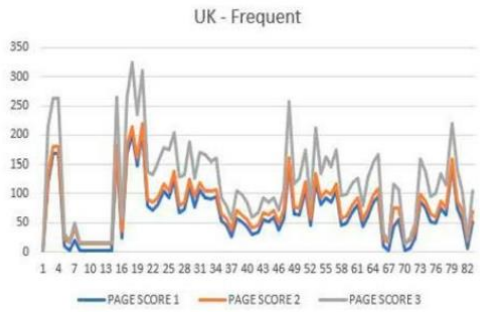

Fig-15: Comparison graph for URLs with their Pagescores for keyword "UK''

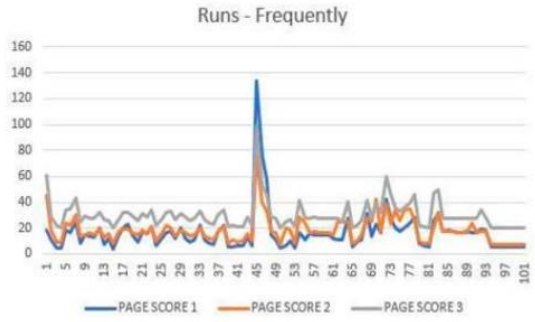

Fig-16: Comparison graph for URLs with their Pagescores for keyword "Runs"

Fig-17: Comparison graph for URLs with their Pagescores for keyword "Android"

5. CONCLUSIONS

- A separate central database system is responsible for maintaining each website's page on the search engine.
- The search engine not only processes user queries but also builds indexes within the database of online pages.
- Web crawlers, often called spiders or robots, are programmes that gather documents from crawled web pages and store them in a web pot. Starting with a seed URL, the crawler's frontier serves as its to-do list. When the crawler visits a website, it eliminates any new links from the documents it has downloaded.
- It verifies whether the user has already downloaded the pages after removing the URL. If the document still hasn't been downloaded, the crawlers will be given the URL to download it again.
- The crawler will keep doing this until it has downloaded every single URL web page. Every single day, crawlers download millions upon millions of online pages. For archiving text and metadata, it has a scheduler, a single-threaded downloader, a multi-threaded downloader, a queue URL, and storage. This component is in charge of maintaining and incorporating the modifications made to the website.
- The bottleneck is addressed by using hierarchical clustering methods, as well as single- and multi-threaded clustering algorithms. It uses the Hypertext Transfer Protocol (HTTP) to get and parse online pages.
- A downloader and processor are two of its parts. In this case, the processor sends pages from the URL process for further processing, while the downloader is utilised to retrieve web pages from the internet.
- A hierarchical clustering approach is used to establish the minimal description length. In this algorithm, all input web page documents are treated as one cluster.
- By analysing their performance in terms of execution and harvest time, we were able to determine the average execution time for the suggested approach.
- These clustering techniques include Text Minimum Description Length (TXT_MDL), Min Hash Jaccard Coefficient, and Min Hash Dice Coefficient.
- When compared to existing algorithms, the suggested one produces superior results.
- To add to that, the three cluster papers have execution times of 150, 300, 450, 702, 1404, 2107, 204, 408, 612, 158.5, 317, and 475.5 for the proposed hierarchical clustering, Minhash Jaccard, and Minhash dice, respectively.
- As a result, the results show that the suggested approach is more accurate than the existing ones.
- Therefore, the suggested approach outperforms the present conventional system in terms of execution time by monitoring harvest and execution time to assess online documents within the context of the website's topology.
- The URL is considered frequently updated if it is changed on a regular basis. It mostly comprises of sites that update the score every second, such as sports news. To show how our crawler handles these types of URLs, we used a single dataset. Runs is the name of the dataset. "Runs" is the user-queried keyword.
- The various datasets may be found at https://www. cricbuzz.com/ parent URL. A URL that does not change its content often or ever is known as a static URL.
- The ANDROID - STATIC dataset, which has the parent URL https://www.andro id.com/, is selected for the Static URL. The ranking is determined by the child URLs acquired from the Android search.
- For each child URL, we determine their page score and then order them accordingly.
- You may make the Crawler search for hidden URLs, such Deep-Web and Dark-Web, to make it work better.
- Adding security measures that stop the crawler from changing the site's content or structure is a great way to make progress.

REFERENCES

[1]. Naresh Kumar, Dhruv Aggarwal (2021).LEARNING-based Focused WEB Crawler. IETE JOURNAL OF RESEARCH https://doi.org/10.1080/03772063.2021.1885312.

[2]. Kumar, M., Bindal, A., Gautam, R., & Bhatia, R. (2018). Keyword query based focused Web crawler. Procedia Computer Science, 125, 584-590.

[3]. Hossen, M. K., Wang, Y., Tariq, H. A., Nyame, G., & Nuhoho, R. E. (2018). Statistical Analysis of Extracted Data from Video Site by Using Web Crawler. In Proceedings of the 2018 International Conference on Computing and Artificial Intelligence (pp. 41-46).

[4]. Jadon, M., Sharma, I., & Sharma, A. K. (2019). Sentiment Analysis for Movies Prediction Using Machine Leaning Techniques. In International Conference on Intelligent Data Communication Technologies and Internet of Things (pp. 457-465). Springer, Cham.

[5]. Mufti, Waseem Akhtar. "ClientNet Cluster an Alternative of Transferring Big Data Files by Use of Mobile Code." World Congress on Services. Springer, Cham, 2019.

[6]. Shamir, R. R., Duchin, Y., Kim, J., Sapiro, G., & Harel, N. (2019). Continuous dice coefficient: a method for evaluating probabilistic segmentations. arXiv preprint arXiv:1906.11031.

[7]. Tanvir, A. M., & Chung, M. (2019). Design and Implementation of Web Cr awler utilizing Unstructur ed data. Journal of Korea Multimedia Society, 22(3), 374-385.

[8]. Choudhary, J., Tomar, D. S., & Singh, D. P. (2019). An Efficient Hybrid User Profile Based Web Search Personalization Through Semantic Crawler. National Academy Science Letters, 42(2), 105-108

[9]. R. Navinkumar, and S. Sureshkumar, "Two-stage Smart crawler for efficiently Harvesting deep-Web Interfaces," International Research Journal of Engineering and Technology (IRJET), Vol. 3, pp. 111–114, 2016. 10.

[10]. Y. Patil, and S. Patil, "Implementation of enhanced web crawler for deep-web Interfaces," International Research Journal of Engineering and Technology (IRJET), Vol. 3, pp. 2088–2092, 2016. 11.

[11]. G. V. Jaybhaye, and A. V. Deorankar, "Machine learning approach for self-adaptive semantic focused crawler based data mining Services," International Journal of Innovative Research in Computer and Communication Engineering, Vol. 4, pp. 2507–2512, 2016. 12.

[12]. P. S. Sekhon, and S. Aggarwal, "Focused web crawling using neural network, decision tree induction and naıve bayes classifier," IJCST, Vol. 5, pp. 155–159, 2014. 13.

[13]. S. Gurav, J. Gilani, V. Gore, and S. Jadhao, "Web content extraction using machine learning," International Research Journal of Engineering and Technology (IRJET), Vol. 5, pp. 4517–4518, 2018. 14.

[14]. D. Taylan, M. Poyraz, S. Akyokus, and M. C. Ganiz, "Intelligent focused crawler: learning which links to crawl," in International Symposium on Innovations in intelligent Systems and Applications (INISTA), Istanbul, 2011, pp. 504–5081.

[15]. Vandana Shrivastava (2017), Meta - Heuristic Approach to Enhance The Performance of Web Crawler, International Research Journal of Engineering and Technology (IRJET), Vol. 10, pp. 604–610.