Journal of Smart Internet of Things (JSIoT)
VOL 2023, No.02 | 10-32 | 2023
DOI: 10.2478/jsiot-2022-0002

Article

Distributed Fundamentals based Conducting the Web Crawling Approaches and Types (Focused, Incremental, Distributed, Parallel, Hidden Web, Form Focused and Breadth First) Crawlers

Aska Ezadeen Mehyadin[1], Lozan M. Abdulrahman[2], Sarkar Hasan Ahmed[3], Riyadh Qashi[4]

[1] Engeering Department, Technical College of Engineering, Duhok Polytechnic University, Duhok, Iraq

[2] Information Technology Management ., Technical College of Adminstration, Duhok Polytechnic University, Duhok, Iraq

[3] Network Department, Technical College of Informatics, Sulaimani Polytechnic University, Sulaimani, Iraq

[4] Vocational School Center 7, Electrical Engineering of the City of Leipzig, Laipzig, Germany

*Corresponding author: (email: sarka.hasan@gmail.com)

Received 10 October 2022; Accepted 27 October 2022; Issue Published 15 December 2022

Abstract: Over the last several years, there has been a significant rise in the number of people getting online and using the internet. Individual hypertext links are available, and any one of them may be used to get access to the resource. There is a variety of hypertext links available. It has been feasible to construct new websites as a result of the growth of crawlers, which has been facilitated by the rise in the number of people who use the internet. Web crawlers are highly evolved search engines that make it simpler for customers to get the information they are searching for on the internet. Web crawlers are also known as web crawlers. In a similar vein, these web crawlers have the potential to be used for more research endeavours in the months and years to come. Furthermore, the information that has been gathered may be used to detect and uncover any connections that are absent, as well as to assess the possibility for expansion inside complicated networks. This can be done by discovering any connections that are missing. The analysis of web crawlers is the primary topic of this study. Topics covered include the architecture of web crawlers, the many types of web crawlers, and the challenges that search engines have while using web crawlers.

Keywords: Distributed Systems; Web Crawling, Crawling Approaches; Crawling Types; Search Engine; WWW.

Copyright © 2023 Journal of Smart Internet of Things (JSIoT) published by Future Science for Digital Publishing and Sciendo . This is an open access article license CC BY (https://creativecommons.org/licenses/by/4.0/)

1. Introduction

Internet is now the most major and important source of information and knowledge that can be found anywhere in the globe. This is true regardless of where you are located. In the first place, the website has to fulfil the fundamental data requirements in order for this to be of any value. The internet has a substantial amount of material that is continuously relevant, and it is feasible to discover this stuff easily. It is possible to find this material on the aforementioned website. It is at this moment in time that the user is provided with queries of this nature [1]. Although the World Wide Web is a massive store of information that may be used to answer a variety of queries, it also includes stuff that is entirely meaningless and should be avoided, despite the fact that it may yield beneficial results. This is the case despite the fact that the World Wide Web may provide helpful answers. In the event that you make use of general search engines like Google, DuckDuckGo, or Bing, it is essential to keep in mind that their standard web crawlers will consistently provide results that may or may not be useful. constantly be ready for this, since it is something that you should constantly be prepared for. Keeping this in mind is something that is really essential to pay attention to. For the purpose of providing an explanation for this phenomenon, it is commonly acknowledged that the World Wide Web is built on a hypertextual interface. The event may be deduced from this framework as the explanation. On the internet, the word "web crawling" is used to refer to the automated process of getting information from the internet by methodically looking through websites in order to acquire content that is relevant and valuable. This process is referred to as "web crawling." By using this method, it is possible to gather information on the links that connect a variety of intriguing websites in an efficient manner [2]. Crawlers are used by search engines in order to collect the websites that are essential to their operations. This is due to the fact that the internet has an enormous amount of websites that are available to users. Due to the fact that they are required to obtain information that is not required in addition to the data that is required, these web crawlers are wasteful in terms of the amount of time, bandwidth, and storage space that they use. The outcomes that they create are both pertinent and irrelevant at the same time. Conventional crawlers use a method that is referred to as breadth-first, which is a strategy[3]. In order to implement this method, it is necessary to do an exhaustive search for all of the child nodes that may originate from a single parent node. To find a solution to the problem that is being explored, this activity is being conducted in order to discover a solution. In order to solve the issue of excessive resource utilisation and the accompanying drop in performance that was brought about as a result of this condition, the classic crawler was subjected to a significant amount of change. The focus of this particular iteration of the project is on topic-specific crawling, with the objective of locating data that is pertinent to the themes that have been discovered. This is done with the purpose of discovering data that is relevant to the topics. As can be seen in the diagram[4] the crawlers are set up to concentrate on a certain portion of the internet rather than searching over the whole of the web. They are able to pull information from the web in a more efficient manner as a result of this. The sections of the internet that web crawlers were instructed to investigate were the only ones that they were permitted to visit without restriction. As a result, their access to the internet was restricted. The function of spiders on the internet, which are more often known as web crawlers, is typically to independently acquire content from the internet. Spiders are utilised for this reason[5]. It is possible that the use of web crawlers will show to be highly beneficial in a substantial number of different situations. By way of instance, they may be used to categorise a significant number of web pages, after then they could be made available to the general public for the purpose of doing information searches. Web crawlers, in contrast to viruses or intelligent agents, do not communicate with one another over the network of linked computers that make up the Internet. Between the two categories of software, this is one of the most significant differences that can be recognised. While on the other hand, it is responsible for initiating requests for content on web servers that were received from a list of sites that was selected in advance [6]. The next paragraphs provide a detailed description of the remaining portions of the article that need to be discussed. An examination and explanation of the operation of a web crawler is presented in the second section of this article. The many different kinds of web crawlers that are now accessible are broken down and explained in this section. Crawling algorithms are going to be the subject of discussion in the fourth section of this installment. A full discussion on the

applications of web crawling is included in Section 5, which can be found here. In Section 6, we will address research that is relevant to the topic at hand, and in Section 7, we will explore the difficulties that are involved with web crawling. This part examines the potential for future advancements in web crawling technologies, which may be found in subsequent sections, including part 8. In addition, the conclusion is presented in portion 9, which is the last portion of the document.

2. Background Theory

- Working of web crawler



Figure 1: Working of Web Crawler

For the purpose of locating fresh and important information that is accessible on the internet, the majority of search engines make use of software programmes that are referred to as spiders. The search engine algorithm relies heavily on spiders as an integral part of its structure. Crawlers, robots, search bots, and simply bots are all names that might be used to refer to these spiders. Crawlers and robots are two more names for each of these creatures. In common parlance, the web crawler that Google employs is often referred to as Google Bot. This is because Google Bot is their official name[7]. The robot, which is often referred to as "crawling the web," is in charge of doing internet searches and gathering information in order to construct a search index for a variety of search engines. This is the primary responsibility of the robot. This is the most important duty that the robot is responsible for. Following the moment at which the software has reached the place where it has started its journey on a particular website, it will proceed to click on each hyperlink that it discovers on the page. It is possible for a large number of computers to simultaneously execute several web crawling programmes, with each server running a separate version of the search engine software[8]. On the other hand, this is a potential that can be realised. The information that is found on a page of a website is saved in a database and may be accessed at a later time. This happens each time a web crawler scans a website. Following the retrieval of a page, the contents of that page are added to the index that is maintained by the search engine. Immediately after the retrieval of the page, this takes place. In this index, you will find the most comprehensive collection of keywords, as well as many samples of those terms that can be found on other websites all over the internet. A web crawl is a process that can be broken down into three primary stages: the first stage involves the setup of the tool, the second stage involves the collection of document

sets, and the third stage involves the operation of the crawler. The first step in the crawling process involves the search bot navigating through the web pages of a variety of different websites. Currently, we are at the first stage of the procedure. The subsequent step, which will be taken after the conclusion of this operation, will be to visit the URLs that have been posted on the website [9]. Subsequent to this, the indexing of the words and content that can be found on the website will take place. When the spider is unable to successfully identify a page that satisfies the requirements for inclusion in the index on a consistent basis, the page is ultimately removed from the index in its entirety. This is the result of the spider's failure to effectively discover the page. In contrast, there are spiders that will do extra checks on the website in order to verify the continuing activity of the web page. These spiders are meant to ensure that the website is functioning properly. The first thing that a spider does when it visits a website is look for a file on the server that is named "robots.txt." After that, it moves on to the next step. One of the first things that a spider searches for is this particular file. Specific instructions for the spider may be found inside this specific file, which you can access by clicking on it. It will be possible for you to establish which aspects of the websites have to be indexed and which sections ought to be ignored if you adhere to these recommendations. Spiders can only be prevented from accessing a robots.txt file by using a robots.txt file, which is the only option available. There is no other method accessible than this one. There are no other methods that have been and continue to be effective. The word "standard" is often used to describe to spiders, despite the fact that they are forced to comply to a specified set of standards. For the sake of consumers, the cooperation between major search engines such as Google and Bing, which is striving towards the adoption of common standards, would be beneficial. The establishment of shared standards would be facilitated by this link[10].

- **Types of Web Crawler**
- Focused Web Crawler
- Incremental Crawler
- Distributed Crawler
- Parallel Crawler
- Hidden Web Crawlers
- Form Focused Crawler
- Breadth First Crawler

- **Focused Web Crawler**

The Crawler is a kind of crawler that is meant to retrieve websites that are linked to one another throughout the internet. Another term for the Crawler is the focused crawler, which is another name for the Crawler. In order for the group to accomplish its goal of developing a repository of knowledge on a certain topic, it is currently looking for materials that are not just specialist but also pertinent to the topic at hand. Given the organisational structure of the website itself, it is common practice to refer to the online news website, which also functions as a topic crawler, as a search engine. This is because the website has a subject crawler. Specifically, this is due to the fact that the website itself functions as a subject crawler. Additionally, in order for the focused crawler to properly perform their task, it is needed for them to identify the following: What characterises the current subject of discussion is the trajectory of significance in the future, which is the issue that is being discussed[11]. At the end of this approach, a decision will be made on the upcoming course of action, and an assessment will also be made regarding the degree of relevance that the page that was presented has to the topic that is now being discussed. Utilising a web crawler that you have targeted comes with a number of advantages that you may take

advantage of. The approach is both economically practical and efficient in accordance with the resources that are available. This is true in terms of the resources that are accessible in terms of both hardware and networks. Additionally, it is scalable, which allows it to significantly reduce the amount of network traffic and downloads. This is a significant advantage. When it comes to search visibility, the use of a web crawler that is especially created for the purpose of crawling the web has the potential to significantly enhance searches [12].

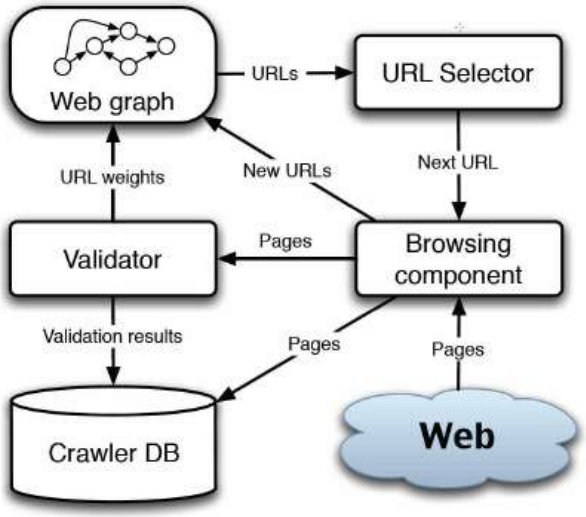


Figure 2: focused crawler architecture

- **Incremental Crawler**

This is done by configuring a typical web crawler to replace older documents with newer ones whenever new documents are uploaded to the website. Particularly at the moment when the new papers are uploaded, this takes place. In contrast to a normal crawler, an incremental crawler visits websites less often but for longer periods of time. This is in contrast to the regular crawler model. The difference between the two is rather considerable in this regard. This technique is based on the understanding that websites are susceptible to changes over the course of time. This is the cornerstone of this strategy. When the programme is in the process of updating its database, it will start by replacing pages that are less significant with new ones that are more vital. The primary goal of this attempt was to find a solution to the issue of freshness, which was a problem that needed to be addressed. One of the many advantages of using an incremental crawler is that it has the ability to selectively relay crucial facts to the user. This is just one of the many advantages. As a consequence, the quantity of network traffic that is used is decreased, and the enrichment of data is enhanced as a result of this[13].

- **Distributed Crawler**

Distributed web crawling is a strategy that may be used in the area of distributed computing. It is one of the ways that can be utilised. A sizeable number of web crawlers are now engaged in the process of actively looking for more websites to include into their crawling protocol. This is being done in order to broaden the scope of the internet that they are able to cover from their current location. Among the numerous nodes that are dispersed around the neighbourhood, one of the servers is located in the middle of the area, and it is the one that is responsible for managing the coordination of data transmission and synchronisation among all of the nodes. It is known as PageRank, and it is the method that is used by the solution in order to deliver a search experience that is more streamlined and better. The distributed

web crawler has the capability of continuing to function on a regular basis even in the event that the system experiences a failure or other unanticipated events take place. An advantage that the distributed web crawler has is this particular benefit. Additionally, it is simple to modify, which enables it to meet the needs of a broad variety of crawling applications. This is a significant advantage[14].

- **Parallel Crawler**

The term "parallel crawlers" is often used to refer to crawlers since it is necessary for many crawlers to be able to perform their functions simultaneously. The purpose of this talk is to define a parallel crawler as a single Process that is comprised of several instances (C-procs) that crawl the internet and are performed on workstations that are linked to a network. In order for the Parallel Crawlers to function in an appropriate manner, it is very essential for them to have a page that is not only new but also the one that they have chosen. It is conceivable for a local network crawler to be physically located at a single site, or it may be dispersed among a number of sites that are geographically distinct from one another. Both of these scenarios are viable. Because of the parallelization of the crawling system, the amount of time it takes to download documents is directly related to the amount of time it takes for this service to complete the download. Specifically, this is due to the fact that the length of time required to obtain the contents of the documents is directly influenced by this feature[15].

- **Hidden Web Crawlers**

By using programming, such as by carrying out a query or filling out an online form, it is only feasible to get access to the information that is kept on the internet. This is the only way to gain access to the information. This is the sole method that can be used at this opportunity. A substantial portion of the information that is stored on the internet is stored in databases, which are responsible for storing this information. In recent years, there has been a significant amount of focus placed on the method of gaining access to a kind of content that is often referred to as the "deep web" or the "hidden web." This emphasis has been dedicated on the process of getting access to this type of content. At the moment, the crawler is focusing its attention on PIW, which is an abbreviation that refers to websites that are available to the general public whether they are located on the internet or on the internet in general. Although it is possible to navigate between these websites by using hyperlinks, search pages and websites that need authentication or prior registration are not included in this category. Other websites that fall into this category include those that are accessible via hyperlinks. It is possible that individuals will choose to ignore a substantial percentage of the very vital information that is disguised by search forms. This is considering the fact that this is something that is taken into consideration.[16]

- **Form Focused Crawler**

If you have online forms that are not widely dispersed, it is likely that the most appropriate option for you to go with is to make use of a Form Focused Crawler. This is because the form focused crawler is designed to focus on the form itself. Utilising a method that enables it to avoid studying pathways that are not fruitful, the Search Form Crawler restricts its research to topics that are associated with search forms. This allows it to avoid wasting time and resources. Because of this, it is able to prevent exploratory inefficiency. In order to do this, it takes use of link components and paths that go to websites that contain types that may be searched. In addition, it is able to perform this while making use of relevant criteria to decide when it is acceptable to stop searching for a certain object[17]. It is possible to view a visual depiction of the process of creating the Form Crawler in the graphic that can be seen below.

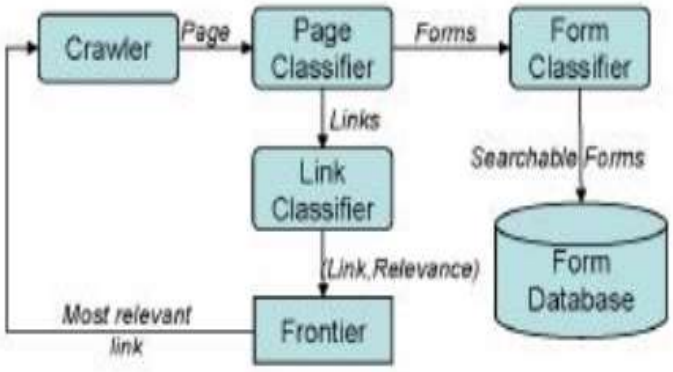


Figure 3: Architecture of Form Focused Crawler

As a means of enhancing the effectiveness of the search procedure, the crawler makes use of both the page classifier and the link classifier. Both of these classifiers are something that the crawler makes advantage of. The final phase involves the use of a third classifier, which is referred to as the form classifier, with the purpose of removing categories that are not necessary. When training the page classifier to classify the pages into the appropriate categories, the taxonomy affiliation of each page is used as part of the training process. This means that it employs the same tactic as the first crawler, which has shown to be the most effective of the bunch.

- **Breadth First Crawler**

First and foremost, the pages begin with a collection of pages that are easier to navigate. This is the case from the very beginning. In order to prepare for the next plan, which will put a larger focus on breadth, this step is completed. Following that, individuals continue to browse by following links in a manner that prioritises the width of the page above the breadth of the page in terms of meaning. It is still possible to construct a variety of alternative traversal algorithms, despite the fact that websites are not often accessed in a manner that is strictly breadth-first. This is because the development of these algorithms is still a viable choice. One example of this would be the likelihood that it would enable prioritising, which would make it possible to search the websites that are the most important first[18].

3. Crawling algorithms

As of now, there are several crawling algorithms to choose from. Before the crawler begins its job, there are a number of simple procedures that are completed:-

- Removing a URL from the URL list.
- Determining the IP address of its host name.
- Downloading of the related documents.
- Extracting any links available in the documents.

4. Uses of web crawling

On the other hand, search engine optimisation (SEO) is only one of the many applications that web crawlers may be used for. Website crawlers may assist a wide variety of applications, and this is just one example of such uses. Furthermore, crawlers may be used to automate regular maintenance tasks on a website, such as analysing links and verifying that the HTML code is valid. This can be accomplished

via the utilisation of crawlers. The use of crawlers to attain this goal is a viable option. Additionally, crawlers are used for a different function in this scenario[19]. For your convenience, the following is a compilation of extra crawler settings that may be used. You can find them below. It is possible to make advantage of the capacity of crawling, which enables the recovery of certain types of data from websites. This capability may be used. As an illustration of this, consider the gathering of email addresses, which is often used for the purpose of sending messages that were not intended to be sent. For the purpose of gathering information from websites that are accessible to the general public on the internet, web crawlers are often used by the most well-known search engines, such as Google[20]. When a visitor arrives at a website with a certain search query, they are shown with webpages that are related to their particular search query. The gathering of data is the means by which this is performed in order to ensure that this takes place. Taking this activity is done with the intention of increasing the speed at which the internet functions. The speed of the internet has grown as a result of this, which is a beneficial outcome that has contributed to the situation. Web crawlers are often used for the aim of doing textual analysis in the field of linguistics, which is concerned with the study of language[21]. At this very minute, searches are being carried out on the internet in order to establish the phrases that are presently being used all over the planet. Crawlers are used rather often in the discipline of biology, especially with the purpose of identifying essential information that is related with a gene that is yet to be found. It is a regular operating practice that this is done. Web crawlers are used by market researchers in order to find and assess trends within a certain market. This is done since market researchers are doing market research. Performing this action is done with the intention of achieving the goal of acquiring knowledge. Through the use of their extensive research, which will be made available to them, they will give guidance on trends that are anticipated to arise in the years to come[22].



Figure 4: Website and application developments

5. Related Work and Sufficient Comparison among them

In this area, we concentrate on studies that have taken a closer look into Web crawling. Information salvaged from the web is acquired through the use of search engines, which retrieve online pages according to the specifications of users. The breadth of the web is massive, encompassing all kinds of data including structured, semi-structural, and non-structured information. This is why search engines need web crawlers. Most of the material online is unmanaged, therefore it is not feasible to access the complete web at once in a single effort, hence search engine users require web crawlers.

The author Shrivastava (2018), who is referred to in reference [23], asserts that search engine crawlers are an important component in the functioning of the whole search engine. Internet browsers are the software programmes that are in charge of surfing the internet and obtaining all of the links that go to websites. Search engines make use of several instances of crawlers that are dispersed throughout a number of servers in order to collect a wide range of information from those servers. This is done in order to enhance the quality of the service that they provide. A web crawler is a piece of software that

navigates the World Wide Web in a systematic fashion, moving from one website to another and then downloading the content of each page it sees. It is a kind of web crawler. When this information is put into the database of the search engine, it is instantly indexed once it has been entered. This index is made up of a comprehensive collection of words and textual content that has been compiled from a range of different platforms throughout the internet. There are many different sorts of web crawlers, and this article provides an in-depth analysis of a web crawler, covering subjects such as its design and the many types of web crawlers. Due to the fact that he has carried out a comprehensive analysis of web crawlers, his study is of great significance. This is because web crawlers that have been created in an efficient manner consistently provide excellent outcomes.

In [24] study that was published in 2018, ALBERTOS-MARCO and colleagues investigate the perspectives of educational users on dispersed encounters in learning environments that include a wide range of device configurations. It is possible for users to connect with their personal computers, tablets, and mobile phones at the same time, which results in the creation of a unified experience across a number of devices. This is done in order to cultivate an educational environment that is enhanced by the technology of the Web interaction hub. A survey was given to a group of more than 150 students in order to ascertain the amount of satisfaction that they were experiencing, and the findings of the poll were then published in a report. As illustrated by their results, which demonstrated that distributed interactions had a favourable affect on pupils, students demonstrated a preference for utilising distributed interactions in multi-device settings rather than traditional interactions. This showed that students preferred using distributed interactions.

The authors of this paper, Baldassarre et al.[25] (2018), analyse whether or not it is possible to apply the ideas presented in the Social Internetworking System to the Internet of Things (IoT). One of the most innovative models that they propose is called MIoT, which is an acronym that stands for Multiple Internets of Things. This model has the capability of effectively replicating and controlling the complex nature of the environment that is associated with the Internet of Things. In addition, they are of the view that the most suitable approach for collecting and using information in the context of a significant number of distinct objects would need the building of a specialised crawler that is designed specifically for an M-I-O-T platform. This is the opinion that they have. It was revealed, as a result of an effort to carry out trials, that the antiquated crawlers that had been used in the past were not suitable for applications that utilised the Internet of Things. One of their new crawlers, on the other hand, turned out to be an outstanding option that beat all of the other choices in terms of performance.

In [26]BUbiNG recently published a publication that includes a presentation of the findings that were acquired by BOLDI et al., 2018. This finding was included in the study. Downloads of the Java application known as BUbiNG are accessible at no cost now. In addition to operating in a distributed form, it is made up of a crawler instance that is capable of crawling hundreds of pages in a single second. Nevertheless, it is essential to have a courteous attitude when crawling a website. This includes demonstrating respect for the host of the website as well as the IP address of the website. Additionally, in contrast to past open-source distributed crawlers, which relied on batch techniques such as MapReduce, BUbiNG takes use of contemporary high-speed protocols in order to optimise throughput, which eventually results in improved performance.

An alternative to batch-based transfer learning is shown in [27]by Navarro1 et al. from 2018. This alternative was developed in 2018. Instead of training each layer on all of the samples from the whole dataset, they provide Weakly Supervised Learning (WSL), which is a two-step transfer learning approach that was designed specifically for deep models. This technique was built more specifically for deep learning. Additionally, the organisation makes use of image-based search in order to increase the

accuracy of its web crawls and address the problem of cross-domain ambiguity inside its online crawls. The WSL library, which was developed by adding the noise structure that was present across classes into their models while they were being trained, made it feasible to selectively extract information from the web data. This was made possible by taking use of the fact that the WSL library was constructed. For the objective of fine-grained skin lesion categorization, they carried out benchmarking with the use of a dataset that was accessible to the general public from the beginning. It was discovered that the implementation of online supervision resulted in a considerable increase in the accuracy of the top-1 classification, which increased from 71.25% to 80.53%. It is possible to attribute this increase in performance to the usage of WSL, which is responsible for the gain in performance.

The research that Lagopoulos and his colleagues [26]conducted in 2018 included the development of an innovative method for identifying internet robots on websites that contain a great deal of content. The researchers came up with this strategy on the basis of the notion that individuals who use the internet have certain activities that they are interested in. This strategy, according to their reasoning, searches through each page in search of links that lead to these particular themes. Additionally, it makes use of two different automated algorithms in order to navigate the internet in a systematic way and identify them. For the purpose of providing a representation of user sessions that incorporates more components, the creators of this programme make improvements to the framework that they already have in place. On the list of these characteristics is a description of the semantics of the resources that the user is being requested to utilise. The indicated semantic qualities have been found to increase the accuracy of online robot detection, based to empirical data that was taken from a website that is linked with a university. This information was obtained from the website.

Klein et al.[28] (2018) undertake a study to determine whether or not it is possible to carry out targeted crawls on the archive web. They are able to efficiently buy 22 web archives that contribute to the overall creation of event collections by using the Memento architecture. This allows them to effectively purchase event collections. They make use of their site collections on four distinct occasions: Valentine's Day, Mother's Day, Father's Day, and Grandfather's Day. This is done with the intention of improving the collection that they have in their library. They analyse the outcomes of a collection that has been meticulously chosen in contrast to the resources that they currently have at their disposal. Based on the findings of their inquiry, it has been determined that not only is it feasible to do targeted crawling of archived websites, but it also has the potential to generate collections that are very relevant, particularly in relation to historical events.

The article [29] by Kumar et al., 2018 presents a proposal for a query-based crawler. When it comes to initiating searches on the respective search interface, this crawler takes use of a list of keywords that are relevant to the user's subject of interest in order to do so. When a seed URL is entered into the search bar of a website, the website will generate search interfaces for websites that are similar to the one that is being searched for and will display them underneath the search field. The crawler will be able to gather new relevant hyperlinks by browsing through the content that is accessible online. This will allow the crawler to avoid exploring any farther into a page. There is currently no way of focused crawling that is capable of making effective use of query-based websites. This is the case at the present time. The job that they provide will show the material that is most pertinent to the keywords that are used in a certain domain. This will be accomplished without the user having to go through a lot of links that are not linked to the issue that is currently being discussed.

A crawler architecture that is suggested by Nakashe and Kolh [30](2018) is one that is made up of a system that is split into two phases. A method that is referred to as "Reverse searching" is used by the Smart crawler throughout the early phase of the procedure. For the purpose of locating a web page that has a URL that is relevant to the query, this method takes use of the query as well as the links that have been found in the site database. This technique, which is known as "incremental prioritisation," is carried

out by the crawler at the second level. It entails comparing the content of the query with the content of the web page. The frequency with which this page appears in search results will be used to assess whether or not it is relevant or irrelevant. This page will then be classed as either relevant or irrelevant. By using a specialised crawler that is designed to carry out targeted searches while taking into consideration the user's personalised professional profile, it is possible to effectively gather information that is relevant to the job market while searching for employment opportunities. The crawler is responsible for carrying out this evaluation via the process of domain categorization. This method supplies the user with information about the quantity of resources that have been made available to search keywords that have been used by the user. There is a separate log file that the crawler maintains in order to keep track of the amount of time that is required for each search operation. This is a wonderful tool for you to use if you want to check search results that give a close match to the phrases that you are now using for your search before you actually enter those keywords into the search box. A focused crawler that is capable of examining the database in an efficient way and giving the user with enhanced results is the key objective of this project. This crawler will be built.

The purpose of Raj et al. [31](2018) was to develop a crawler that is constructed using distributed components and is intended for use with deterministic AJAX apps. The reduction of the expansion of state space, the development of time efficiency, and the supply of comprehensive content coverage were placed at the forefront of their goals. To accomplish the task of delivering the response, this strategy employs a variety of diverse tactics in conjunction with one another. Furthermore, it is of the highest significance to take notice of the fact that this strategy, which is dictated by components, helps to the decrease of the expansion of the state space. During the second stage, a Distributed-Crawling approach is used in order to simultaneously manage the occurrences with the objective of enhancing the efficiency of the organisation. An strategy known as Breadth-First Search (BFS) is used by the system in order to provide comprehensive coverage of the material in order to achieve the aim of giving wide coverage.

The research conducted by Aggarwal [32]in 2019 contains an in-depth analysis of a variety of focused crawling algorithms. Each of these strategies employs a unique set of criteria to assess the benefits and drawbacks of predicting the relevancy of URLs. The purpose of this study is to give a comprehensive examination of the different techniques. The researchers develop the issue formulation after performing an investigation into the previously done study on targeted crawlers. This involves explaining the problem and coming up with a remedy to the problem. These steps are taken after the researchers have completed the examination.

In the course of designing web tests, Biagiola et al. [33](2019) came up with a novel method to the process. By using this method, future test cases are prioritised in accordance with the diversity of the tests that are currently in place, which ultimately leads in an improvement in flexibility. Therefore, the only test cases that are taken into account for execution inside the web browser are those that examine a broad range of application behaviours. This is because the web browser is the only place where this is possible. One of the products that the firm has produced in order to put its notion into action is a piece of software that is known as DIG. Their empirical investigation of six real-world online applications revealed that DIG outperforms other web test generators, including crawling-based and search-based generators, in terms of coverage and early defect diagnosis rates. This was the conclusion reached by the researchers.

For the aim of data collection, Koloveas et al.(2019) provided a design that employs a two-phase process [34]. This architecture was shown to be effective. Immediately after the conclusion of the first phase of the harvesting technique, a crawler that is powered by machine learning is used to guide the harvest in the direction of the websites that are desired. During the second stage of the process, advanced statistical language analysis techniques are used in order to display the information that has been acquired in a hidden, reduced-dimensional feature space and to prioritise it according to the possible significance it may have for the job that is now being performed. For the goal of proving that the proposed design is possible, it is only carried out through the utilisation of open-source technology, and

the first evaluation helps to validate the effectiveness that was considered to be anticipated.

An investigation was carried out by Mehak et al. [35](2019) in which they used web crawling and scraping techniques in order to ascertain whether or not five distinct e-commerce websites provided the best deals. HTML and CSS are employed for the front-end of the framework, while PHP is utilised for the back-end of the framework. This allows for the framework to be constructed. The Hypertext Preprocessor, or PHP, is used for the purpose of controlling the front-end components that are produced using HTML and CSS. This is accomplished via the utilisation of PHP. Python modules are used in the process of developing scraping apps, and online crawls are utilised in order to get information about web labels. The peculiarity is in the fact that their system does not collect data for information that has already been gathered and stored in the local database. This particular aspect of their system is one of its characteristics. Instead, data is gathered in real time whenever it is required, and it is shown immediately after a search is performed whenever it is required. The data is requested, which causes something to take place. The ability to appropriately manage and optimise the storage and processing capacities, as well as the capacity to optimise them, will be improved as a result of this. Additionally, the method for extracting data achieves an accuracy of 93%, which is achieved with a low amount of computation and time requirements. This is done using the methodology.

In the study that Nasr et al.[36]carried out and eventually published in 2019, their purpose was to analyse the links between entities by making use of elements other than weights, namely the ITF-IDF. Rather of parsing HTML, this method gathers information by making use of visual signals that are present on online pages. In contrast to this, code-based web page analysis is being discussed here. Through the use of a layout tree, it is possible to get all of the records. The Noise Filter (NSFilter) algorithm is used in the design that they have built in order to eliminate components that are not required. These components include headers, footers, advertisements, and other information that is not requested. Having chunks that are visually distinct and being structured in a way that is comparable are two characteristics that distinguish data records from other types of records. Instead of classifying the visual blocks according to colour, layout, and shape, the idea suggests clustering them based on the similarity in form and coordinate feature (SSC), which is a mix of similarity in appearance and similarity in form. This is an alternative to the traditional method of visually categorising the blocks. The findings of the experiment indicate that the framework that was proposed is superior to the data extraction methods that were previously used, as was previously claimed.

Nathezhtha and his colleagues[37] introduced a method that they called the Web Crawler based Phishing Attack Detector (WC-PAD) in their research paper that was published in 2019. This method is intended to accurately detect phishing attacks in three separate phases. In order to categorise websites as either phishing or non-phishing, the inputs include of websites that receive internet traffic, online content, and URLs. Additionally, the characteristics of these websites that are often used to categorise them as either phishing or non-phishing websites are also included in the inputs. The experimental examination of the WC-PAD is one of the most cutting-edge methods that may be used to determine the level of competence achieved by the WC-PAD. According to what was said before, it makes use of datasets that were compiled via in-depth research on actual instances of phishing. Upon doing an analysis of the results of the testing, it was found that the WC-PAD that was proposed works very well in recognising phishing attacks as well as zero-day phishing attacks, with an accuracy rate of 98.9%. This was determined after the results of the tests were analysed.

An enhancement to SUMMON was described by Pflanzner et al.[38] (2019), and the authors of the study referred to it as the Automatic Web Crawling Service (AWCS). The objective of this service is to conduct searches on websites that are open to the general public in order to locate data pertaining to the Internet of Things (IoT) and sensors. This introduction is meant to provide a general understanding of the structure and operation of this approach, which is the objective of this introduction. This is shown via the use of three different situations that occur in the actual world. It is possible for simulators to take use of the archiving approach that is provided in order to carry out testing that is accurate.

Zhu et al. (2019) [39] conducted an analysis on a variety of various bandwidth allocation techniques for a web crawler system. This investigation was carried out throughout the course of 2018. The architecture of this system has been designed to guarantee that it is able to perform well in a broad range of different situations. The scenarios that fall under this category include those in which the sequence of website crawling cannot be altered, as well as those in which it is possible to make adjustments to the sequence. During the first scenario, they provide several methods that may be used to restrict the amount of bandwidth that web crawlers have access to. The goal of these strategies is to either cut down on the maximum amount of time that web crawlers are required to complete their jobs or to cut down on the total amount of time that each web crawler has to go through in order to do their work independently. Whenever crawling priority is utilised in order to create greedy values, there are two greedy algorithms that may be utilised. Both of these algorithms are available for usage. The basis of one of these algorithms is crawling priorities, while the other is based on mixed-integer programming. Both of these algorithms are implemented respectively. A number of extensive simulations are carried out in order to verify the recommended approaches. The results of these simulations demonstrate that the processes provide beneficial performance, which is the conclusion that can be drawn from the simulations.

Prusty et al. (2020) in reference [40] argue that because to the complex structure of the internet, it is inefficient for "single" crawlers to effectively crawl and index the content of the internet. This is because of the nature of the internet itself. Since this is the case, a substantial portion of internet users who are unable to crawl the web via the application of programming methods and who do not possess the technical expertise required to do so may resort to using third-party crawling services in order to do this task. Using containerisation technologies such as Docker and Kubernetes, the application is largely focused on finding a solution to the problem of scalability and performance at the same time. This is done in order to increase both the large-scale performance and the performance overall. There is a possibility that the system will be simple to access if it is designed using a user interface that is based on the web. As an alternative to employing a single instance that is referred to as "manager," they have made the decision to use several containerised Crawler-Manager instances for each and every user request that is received by the application. Machine learning may be used to minimise the amount of storage space that is needed; this is performed by selectively preserving data that is classified according to user-defined class labels. This is accomplished by using machine learning. That is to say, they compared our strategy to the efficiency of the parallel crawling tactics that are now being used. In other words, they assessed the effectiveness of the strategies.

While conducting their study, Pushpa C. N.[41] an innovative approach to the discovery of online services that takes semantic awareness into account was offered by the author and colleagues in the year 2020. In addition, the technique in which they utilise is the most appropriate for gaining the semantics that lie under the surface, and it also provides clients with the capacity to make use of services that are offered online. The objective of the plan that is now being implemented is to develop a web-based database that will include all of the datasets that have been identified as having originated from membership registration websites. Utilising clustering that is based on NPMI values may be performed via the use of word-level semantics. The semantic similarity values are the basis upon which the library is constructed via the construction process. Establishing a taxonomy for the primary classifications of web services is done in order to make clustering more straightforward. The initial collection of information and concepts served as the foundation for this taxonomy. Web service discovery is a method that has been successfully established in order to enable the offering of relevant online services to customers based on the search queries that they have entered. A significant leap forward in comparison to the performance of techniques that were taken into consideration in the past is the achievement of an accuracy rate of 90.4%.

Biagiola and his colleagues [42]offered a technique that they called DANTE in their research paper that was released in the year 2020. The generation of executable test schedules is accomplished by the use of a web test generator using this way. The test dependencies of the test cases that were obtained

during a crawling session are taken into account by this generator, which was designed specifically for the purpose of making these schedules. This generator was developed in order to produce these schedules. As a result of the development of programming tools by DANTE, it is now feasible to transform a web crawler into a test case generator. It is the duty of this generator to construct minimal test suites that are made up of just those tests that are feasible and are able to effectively achieve the coverage that has been defined. The results of a series of experiments demonstrate that DANTE is capable of accomplishing the following: (1) it reduces the error rate of test cases obtained through crawling traces from 85% to zero; (2) it generates smaller test suites compared to the initial ones, which results in a reduction in overall size; and (3) it outperforms two competing techniques, one of which is based on crawling and the other of which is based on modelling, in terms of coverage and breakage rate.

Researchers are able to estimate the parameters of the unknown model by making use of only fragmentary data, as stated by Upadhyay et al.[43] (2020). Specifically, this is because researchers believe that they are only able to analyse change rates in real time if they are aware of the parameters beforehand. This is the reason why this is the case. To be more exact, they employ single-bit signals to detect whether or not the page has been changed since the last time it was refreshed. This is done in order to determine whether or not the page has been updated. Providing an explanation of the specific situations in which solutions may be identified is something that the authors do in order to facilitate a more in-depth grasp of the circumstances in which partial observability leads in solutions. Following that, they propose a viable estimate and produce confidence ranges for it based on the length of time that has elapsed between the pieces of data. When it comes down to it, the researchers have shown that the explore-and-commit strategy leads to regret of $O(\sqrt{T})$, and this regret is reduced when a more conservative exploration horizon is used. The simulation study that was carried out demonstrates that our online technique is able to scale fast, and it provides performance that is extremely close to the ideal for a broad variety of parameters. This was shown by the fact that it delivers it.

The purpose of the present research, which was carried out by Arillotta et al. [44](2020), is to conduct an analysis of the different online platforms and forums that are used by individuals who are interested in exploring the use of opioids within the context of the internet environment. Following that, the programme discovered a total of 426 opioids, of which 234 were analogues of fentanyl. This was discovered by the software that crawls and navigates the open web. In a subsequent experiment, this was shown to be the case. Even up until quite recently, the great majority of these cutting-edge and emerging compounds were mostly unknown to the general public. However, there is a lack of docking, preclinical, and clinical research on these analogues, which is a reason for concern. The fact that each of these analogues has the potential to demonstrate diverse toxicodynamic profiles is a matter for concern. One strategy that might be utilised to enhance the way that the public health issues that are associated with opioids are explored is to increase the amount of multidisciplinary cooperation that takes place between medical experts and bioinformatics.

During the course of the study that they carried out, Avrachenkov and his colleagues [45](2020) presented three novel methods for determining the pace at which web pages are subject to modifications in real time. Because each of these tactics needs only a partial understanding of the process of page change, namely the determination of whether or not the page has been modified since the last crawl instance, each of these strategies is possible. This is because each of these strategies requires only little knowledge of the process. The first tactic is predicated on the notion that large numbers are more significant than low numbers that are lower. The second strategy, which is based on the concept of stochastic approximation, is used in order to achieve the goal of improving the outcomes. In addition, the third technique contains an additional momentum component, which enables it to build upon the results that were acquired via the use of the second method. In each and every one of these projects, the proof and convergence rates for each of these approaches are shown to the user. According to their understanding, the findings that they have arrived at about the third estimator are rather distinctive. A stochastic approximation approach that utilises momentum has obtained its first convergence conclusion

using this specific technique. At the end of the day, the authors perform a large number of numerical experiments using both real and synthetic data in order to assess the usefulness of their recommended estimators in comparison to current methodologies such as Maximum Likelihood Estimation (MLE).

In their paper [46], Jakubicek et al., 2020 highlight the issues they had when building corpora using Sketch Engine, a modern tool for doing so. Also included in their talk are the challenges they experienced. Instead of supplying the audience with preconceived ideas, the objective of their post is to offer a complete appraisal of viable responses and to promote critical thinking among the readers. In most circumstances, they will assess each problem, establish the degree of its severity, and then explore the numerous approaches in which the harm it produces may be minimised.

One of the research papers that Shamrat et al. [47](2020) presented in their publication was a research study that discussed the building and execution of a web crawler. The objective of this web crawler was to take information from the internet and then use data filtering procedures in order to make the content more accessible and visually attractive for end users.

A unique concept that they name "Crawler by Inference" is proposed by Hegade et al. [48](2020) in order to accomplish the goals that they have outlined. The rules of inference, semantic similarity, and paradigmatic similarity are all employed in the construction of this theory. An new strategy was devised that provides priority to connections based on the amount of rules that have been found or generated in the most recent time period. A model is provided by the researchers, which postulates the existence of an intelligent queue, which is a data structure that is both efficient and effective. The connections in this queue are prioritised according to the significance of their status. It is possible that the data of a page, in conjunction with the conclusions of the analysis, might serve as meta-data for the page. In addition, the tables that compare their results to those of a normal crawler are given in the paper that they have produced. By lowering the number of irrelevant sites that need to be indexed, their method works towards the goal of improving the results.

UZUN[49] presents a revolutionary method that he calls UzunExt in the essay that he wrote for the year 2020. The text is extracted using string techniques, and additional information is also included. This method does not need the creation of a DOM Tree, and it also includes the inclusion of supplemental information. In order to complete the process, you will first search for a pattern in a sequential manner, then count the number of closing HTML elements that are linked with that pattern, and then extract the content. It is possible for them to gather extra information by beginning the crawling process. This information may include the starting position, which can be used to optimise the search process, the number of inner tags that are required for that process, and the amount of tag repetition that is required to finish the process. With regard to the DOM-based approach for extracting strings, the algorithms that are related with string manipulation in this specific methodology provide a speed advantage that is around sixty times greater than the latter. A further advantage of using this supplemental data is that it reduces the amount of time necessary for extraction by a factor of 2.35 when compared to the use of string approaches alone. In addition, he makes use of a method that is very flexible and capable of being simply implemented by other DOM-based researchers and parsers in order to improve the efficiency with which they utilise their time.

In their research that was released in the year 2020, WANG and colleagues [50]presented JWASA, which provides a great deal additional features. The capacity to automatically construct ontologies or vocabularies, the ability to extract functional semantics in an automated way, and the ability to add metadata to your data are all examples of these capabilities. Additionally, JWASA offers a system for the generation of bridge rules that is established in a semi-automatic manner. Using this method, it is possible to infer inferred connections between different languages, such as sub-class, super-class, and equality links. One of the key objectives of JWASA is to perform semantic annotations on the functionality of Web APIs. This will allow for the creation of useful semantic Web APIs that can be used for future API development. In order to test the effectiveness of JWASA on genuine Internet Web

APIs that are collected via web crawling, they first develop a basic version of the system and then carry out a series of experiments. This assumption is supported by a number of trials, which demonstrate that their method is not only successful but also efficient.

According to the findings of their research publication [51], Xu et al. (2020) suggest using a distributed crawler strategy to collect data pertaining to cities. Moreover, several cities have shown that they are capable of being crawled in an effective manner. Processing, data analysis, and several other methods are used to preliminary data, which includes data gathered via web crawling, in order to successfully extract information that is pertinent to the situation. The facts make it abundantly evident that the strategy that has been offered will be of great assistance to the administration of the municipality.

Table1: Comparison among most related previous works for Web Crawling Approach

Ref.	The Proposition	Methods	Their Results
[23]	Researchers conducted a study that investigated how people perceive dispersed interactions.	Search engines	It usually provides successful outcomes about the majority of the time
[24]	This presentation offers a full analysis of a web crawler, along with a number of distinct designs	Multi-device setups	The results show the impact that distributed interactions made on students
[25]	The Social Internetworking System, which was developed by Vinton Cerf and colleagues in 2010, is being considered as a way to implement IoT.	MIoT (Multiple Internets of Things)	It is shown that conventional crawlers are not sufficient for mobile internet of
[26]	They provide a unique online robot identification methodology for websites that are rich in information.	Web robot detection	The proof is provided by an academic publisher, which concludes that semantic properties increase the accuracy of web robot identification.
[27]	They consider how feasible it would be to execute concentrated crawls on the archived web. In the event that using the Memento infrastructure is necessary.	Memento infrastructure	Based on their findings, it can be concluded that targeted crawling of archived online content may be done and produces meaningful collections.
[28]	The crawler operates in two stages. These are the Reverse searching and Incremental prioritization stages.	E-learning application	They are using an E-learning program, and so, they have the capability to improve their efficiency in terms of online searches as well as teaching individuals.
[29]	A distributed	Concept	Using a Breadth-First

	component-based crawler that works with deterministic AJAX apps should be developed.	of Machine Learning	Search (BFS) technique, they help ensure that they give full material coverage
[30]	This technique provides a revolutionary online test generating method that picks the most promising test cases to be presented	DIG (DIversity based Generator)	DIG's coverage and fault detection rates were increased substantially.
[31]	It is important to focus on the information collecting assignment.	Machine learning-based crawler and State-of-the-art statistical language	Natural language understanding is used to determine if there are entities
[32]	The five e-commerce websites that were identified using a combination of web crawling and scraping were researched using these approaches.	HTML, CSS, PHP and python libraries	It helps to improve the storage and processing ability as well as the data retrieval process accuracy, which is 93% with the least amount of calculation and time.
[33]	A new architecture for the efficient search of deep web data records is suggested.	Layout tree and Noise Filter (NSFilter) algorithm	The findings from the experiment indicate that the framework being suggested is superior to previously documented data extraction approaches.
[34]	a Phishing Attack Detector (WC-PAD) has been suggested and a Uniform Resource Locator (URL)	DNS Blacklist and Web Crawler based detection	Accuracy was tested on a WC-PAD and it found that it provided a detection accuracy of 98.9% against both phishing and zero-day phishing attacks.
[35]	Approached a solution to the problem by means of a method utilizing distinct limits on how quickly a crawler system may scan the Internet.	A crawling-prioritized method implemented using integer programming and an integer-programming-based optimization methodology	Prove that their method gets the job done in a wide range of circumstances
[36]	This particular	Django,	instead of having a single

	solution has these strengths and attributes as it is a horizontally scalable web crawler that makes use of containerization (Docker and Kubernetes) for scalability and performance.	Docker, Kubernetes and Python	manager instance that manages several crawler instances, the service focuses on creating several instances of the Crawler-Manager for each incoming user request
[37]	The introduction of a novel web service discovery technique based on semantics is proposed.	Web database	They want to apply a strategy that achieves an accuracy of 90.4 per cent, which is far better than the methods they now use.
[38]	The authors introduce a revolutionary technique to web test creation, which is implemented in a program named DANTE (Dependency-Aware Crawling-Based Web Test Generator)	Web Testing Techniques	Using DANTE's test suites, their experiments have found that test suites generated by DANTE are on average 84% smaller than the ones created by the crawler.
[39]	(Conceived, built, and coded) a web crawler to collect information from the internet and process it in such a way that the results are usable and graphical for the user	Python Programming Language	This means that read-write and stored functions must be obtained before the actual python raw code is run.
[40]	They have come up with a fresh idea for a Crawler — named Inference Crawler by semantic similarity, paradigmatic similarity, and rules of inference.	Semantics	According to their model, they claim to provide better results by avoiding the trap of useless sites.
[41]	pushes for an innovative strategy, known as UzunExt, which extracts content using string techniques and additional metadata and does not go through the tedious process of establishing a DOM tree.	Data Collection, Preprocessing, Analysis, and data Visualization	Because of how simple it is to adapt to various DOM-based studies/parsers in this assignment, time efficiency has been enhanced.
[42]	The crawler uses a distributed architecture, which they propose, and gathers city-related data with the architecture in	data preprocessing	Demonstrate that the given technique will be extremely beneficial for municipal managers

	place.		

6. Challenges in Web Crawling

- **Scale**

Web expansion is proceeding at an accelerating rate day by day. For the crawlers to be able to cover a vast area and provide decent performance, it has to have a high throughput. The vast number of engineering-based difficulties that followed from this caused the birth of the profession of structural engineering. In order to overcome these issues, corporations will have to use several computers, each of which may be numbered in the thousands, and more than a dozen high-speed network lines.

- **Content Selection Tradeoff**

At the moment, we have to settle for smaller crawlers that give fast web crawling throughput, but are unable to crawl the complete web or keep up with the ever-changing internet. Crawling is an attempt to gather higher-value material with greater speed, as well as gather all the possible information in a place where it can be located. To provide a clean, streamlined user experience, crawlers should discard any unnecessary, redundant, and harmful information.

- **Social Obligations**

It is important that crawlers follow safety protocols in order to avoid a denial-of-service attack. Crawlers should operate along with the many websites they're assigned to, so they may effectively carry out their tasks.

- **Adversaries**

There are certain content providers who try to introduce unnecessary stuff into the corpus of the crawler. The motives for running such sorts of operations include the desire for monetary gain like misdirecting Traffic to commercial websites[38].

- **Copyright**

From the definition, we may deduce that crawlers presumably do criminal acts by making permanent copies of copyrighted content (web pages) without authorization. Copyright is without question the most critical legal matter affecting search engines. As it strives to save as many web pages as possible, the Internet Archive has found it especially difficult to keep up with the number of web pages stored and made available online.

- **Privacy**

The problem of personal privacy is a lot clearer for crawlers as everything on the web is in the public domain. While web information may still invade privacy when it is employed in certain ways, such as when the information is aggregated on a broad scale across multiple online sites, there are other instances in which web information does not infringe privacy.

- **Cost**

A web crawler may cost the owners of the websites that it crawls additional money due to having used up their bandwidth allowance. There are many different web hosts, each offering a particular type of server services and charging in a different method. A broad variety of bandwidth options are offered by different hosts. The cost of going over the bandwidth capacity leads to an increase in bandwidth costs, which causes an additional cost to be paid for having the website deactivated.

7. **Future scope of web crawling**

The already large quantity of research that has been done in the area of online data extraction methods has just been multiplied. In the future, the effectiveness of algorithm implementation may be

improved by incorporating newer, more efficient ways. In addition to the current search engine functions, new search engine features may potentially be incorporated, including quicker and more accurate search results. It is essential that the various crawling algorithms' work be enhanced in order to raise web crawling speed and accuracy.[39]. There is a vast amount of study that still needs to be done on this important issue, especially with regard to the scalability of the system and the different processes employed by its constituent components. The ultimate goal is to build a testbed that contains multiple workstations, each with a duplicate copy of the web, in order to make this happen. This testbed is flexible and enables the modeling of the whole web, using either artificially constructed pages or a stored partial snapshot of the web.[40].

8. Conclusion

Crawlers allow search engines like Google and Bing to improve their crawling skills. Crawlers are used to locate and get information from websites. via the use of their components, web services make use of the components that they offer as the essential building blocks for all web services. This is done in order to reach the highest possible level of performance via the utilisation of their components. Web crawlers are computer programmes that have the potential to modify data in a wide variety of different ways thanks to their unique capabilities. The construction of a web crawler programme that is capable of carrying out a variety of tasks is not a difficult procedure by any stretch of the imagination. One reason for this is because the procedure is easy to understand. With this in mind, it is of the highest significance to make use of appropriate techniques and to design a sturdy structure in order to ensure that the programme has a high level of intelligence when it is implemented. Internet search engines use a broad variety of different crawling algorithms to get results from the internet. Make use of an automated crawling algorithm that has been updated in order to achieve one's goals of providing improved results and gaining increased overall efficiency. This is important in order to achieve one's goals.

References

[1] M. M. Sadeeq, N. M. Abdulkareem, S. R. Zeebaree, D. M. Ahmed, A. S. Sami, and R. R. Zebari, "IoT and Cloud computing issues, challenges and opportunities: A review," Qubahan Academic Journal, vol. 1, no. 2, pp. 1-7, 2021.

[2] S. R. Zeebaree, H. M. Shukur, L. M. Haji, R. R. Zebari, K. Jacksi, and S. M. Abas, "Characteristics and analysis of hadoop distributed systems," Technology Reports of Kansai University, vol. 62, no. 4, pp. 1555-1564, 2020.

[3] P. Y. Abdullah, S. Zeebaree, K. Jacksi, and R. R. Zeabri, "An hrm system for small and medium enterprises (sme) s based on cloud computing technology," International Journal of Research-GRANTHAALAYAH, vol. 8, no. 8, pp. 56-64, 2020.

[4] J. Saeed and S. Zeebaree, "Skin lesion classification based on deep convolutional neural networks architectures," Journal of Applied Science and Technology Trends, vol. 2, no. 01, pp. 41-51, 2021.

[5] P. Y. Abdullah, S. Zeebaree, H. M. Shukur, and K. Jacksi, "HRM system using cloud computing for Small and Medium Enterprises (SMEs)," Technology Reports of Kansai University, vol. 62, no. 04, p. 04, 2020.

[6] S. Zeebaree, R. R. Zebari, K. Jacksi, and D. A. Hasan, "Security approaches for integrated enterprise systems performance: A Review," Int. J. Sci. Technol. Res, vol. 8, no. 12, pp. 2485-2489, 2019.

[7] N. O. Salim, S. R. Zeebaree, M. A. Sadeeq, A. Radie, H. M. Shukur, and Z. N. Rashid, "Study for food recognition system using deep learning," in Journal of Physics: Conference Series, 2021, vol. 1963, no. 1: IOP Publishing, p. 012014.

[8] K. Jacksi, R. K. Ibrahim, S. R. Zeebaree, R. R. Zebari, and M. A. Sadeeq, "Clustering documents based on semantic similarity using HAC and K-mean algorithms," in 2020 International Conference on Advanced Science and Engineering (ICOASE), 2020: IEEE, pp. 205-210.

[9] R. R. Zebari, S. R. Zeebaree, Z. N. Rashid, H. M. Shukur, A. Alkhayyat, and M. A. Sadeeq, "A Review on Automation Artificial Neural Networks based on Evolutionary Algorithms," in 2021 14th International Conference on Developments in eSystems Engineering (DeSE), 2021: IEEE, pp. 235-240.

[10] S. A. Mostafa et al., "Applying Trajectory Tracking and Positioning Techniques for Real-time Autonomous Flight Performance Assessment of UAV Systems," Journal of Southwest Jiaotong University, vol. 54, no. 3, 2019.

[11] Z. S. Hammed, S. Y. Ameen, and S. R. Zeebaree, "Investigation of 5G wireless communication with dust and sand storms," Journal of Communications, vol. 18, no. 1, 2023.

[12] L. M. Abdulrahman, S. R. Zeebaree, and N. Omar, "State of Art Survey for Designing and Implementing Regional Tourism Web based Systems," Academic Journal of Nawroz University, vol. 11, no. 3, pp. 100-112, 2022.

[13] R. K. Ibrahim et al., "Clustering Document based on Semantic Similarity Using Graph Base Spectral Algorithm," in 2022 5th International Conference on Engineering Technology and its Applications (IICETA), 2022: IEEE, pp. 254-259.

[14] N. M. ABDULKAREEM and S. R. ZEEBAREE, "OPTIMIZATION OF LOAD BALANCING ALGORITHMS TO DEAL WITH DDOS ATTACKS USING WHALE OPTIMIZATION ALGORITHM," Journal of Duhok University, vol. 25, no. 2, pp. 65-85, 2022.

[15] R. K. Ibrahim, S. R. Zeebaree, K. Jacksi, M. A. Sadeeq, H. M. Shukur, and A. Alkhayyat, "Clustering document based semantic similarity system using TFIDF and k-mean," in 2021 International Conference on Advanced Computer Applications (ACA), 2021: IEEE, pp. 28-33.

[16] B. W. Salim and S. R. Zeebaree, "Kurdish Sign Language Recognition Based on Transfer Learning," International Journal of Intelligent Systems and Applications in Engineering, vol. 11, no. 6s, pp. 232-245, 2023.

[17] H. M. Zangana and S. R. Zeebaree, "Distributed Systems for Artificial Intelligence in Cloud Computing: A Review of AI-Powered Applications and Services," International Journal of Informatics, Information System and Computer Engineering (INJIISCOM), vol. 5, no. 1, pp. 1-20, 2024.

[18] S. Mohsin, B. W. Salim, A. K. Mohamedsaeed, B. F. Ibrahim, and S. R. Zeebaree, "American Sign Language Recognition Based on Transfer Learning Algorithms," International Journal of Intelligent Systems and Applications in Engineering, vol. 12, no. 5s, pp. 390-399, 2024.

[19] L. M. Abdulrahman, S. H. Ahmed, Z. N. Rashid, Y. S. Jghef, T. M. Ghazi, and U. H. Jader, "Web Phishing Detection Using Web Crawling, Cloud Infrastructure and Deep Learning Framework," Journal of Applied Science and Technology Trends, vol. 4, no. 01, pp. 54-71, 2023.

[20] A. A. Yazdeen, R. Qashi, H. S. Malallah, L. M. Abdulrahman, and M. A. Omer, "Internet of Things Impact on Web Technology and Enterprise Systems," Journal of Applied Science and Technology Trends, vol. 4, no. 01, pp. 19-33, 2023.

[21] H. S. Malallah, R. Qashi, L. M. Abdulrahman, M. A. Omer, and A. A. Yazdeen, "Performance Analysis of Enterprise Cloud Computing: A Review," Journal of Applied Science and Technology Trends, vol. 4, no. 01, pp. 01-12, 2023.

[22] M. A. Omer, A. A. Yazdeen, H. S. Malallah, and L. M. Abdulrahman, "A Survey on Cloud Security: Concepts, Types, Limitations, and Challenges," Journal of Applied Science and

Technology Trends, vol. 3, no. 02, pp. 47-57, 2022.

[23] V. Shrivastava, "A methodical study of web crawler," Journal of Engineering Research and Application, vol. 8, no. 11, pp. 1-8, 2018.

[24] F. Albertos-Marco, J. A. Gallud, and V. M. Penichet, "A study on how users perceive distributed interactions on web applications," IEEE access, vol. 6, pp. 57389-57399, 2018.

[25] G. Baldassarre, P. L. Giudice, L. Musarella, and D. Ursino, "The MIoT paradigm: Main features and an "ad-hoc" crawler," Future Generation Computer Systems, vol. 92, pp. 29-42, 2019.

[26] A. Lagopoulos, G. Tsoumakas, and G. Papadopoulos, "Web robot detection: A semantic approach," in 2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI), 2018: IEEE, pp. 968-974.

[27] F. Navarro, S. Conjeti, F. Tombari, and N. Navab, "Webly supervised learning for skin lesion classification," in Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II 11, 2018: Springer, pp. 398-406.

[28] M. Klein, L. Balakireva, and H. Van de Sompel, "Focused crawl of web archives to build event collections," in Proceedings of the 10th ACM Conference on Web Science, 2018, pp. 333-342.

[29] M. Kumar, A. Bindal, R. Gautam, and R. Bhatia, "Keyword query based focused Web crawler," Procedia Computer Science, vol. 125, pp. 584-590, 2018.

[30] S. M. Nakashe and K. R. Kolhe, "Smart approach to crawl web interfaces using a two stage framework of crawler," in 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018: IEEE, pp. 1-6.

[31] S. Raj, R. Krishna, and A. Nayak, "Distributed component-based crawler for AJAX applications," in 2018 Second International Conference on Advances in Electronics, Computers and Communications (ICAECC), 2018: IEEE, pp. 1-6.

[32] K. Aggarwal, "An efficient focused web crawling approach," in Software Engineering: Proceedings of CSI 2015, 2019: Springer, pp. 131-138.

[33] M. Biagiola, A. Stocco, F. Ricca, and P. Tonella, "Diversity-based web test generation," in Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, 2019, pp. 142-153.

[34] P. Koloveas, T. Chantzios, C. Tryfonopoulos, and S. Skiadopoulos, "A crawler architecture for harvesting the clear, social, and dark web for IoT-related cyber-threat intelligence," in 2019 IEEE World Congress on Services (SERVICES), 2019, vol. 2642: IEEE, pp. 3-8.

[35] S. Mehak, R. Zafar, S. Aslam, and S. M. Bhatti, "Exploiting filtering approach with web scrapping for smart online shopping: Penny wise: A wise tool for online shopping," in 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), 2019: IEEE, pp. 1-5.

[36] M. Nasr, H. Fahmy, and M. Thabet, "Efficiency Improvement Approach of Deep Web Data Extraction," in 2019 14th International Conference on Computer Engineering and Systems (ICCES), 2019: IEEE, pp. 214-221.

[37] T. Nathezhtha, D. Sangeetha, and V. Vaidehi, "WC-PAD: web crawling based phishing attack detection," in 2019 International Carnahan Conference on Security Technology (ICCST), 2019: IEEE, pp. 1-6.

[38] T. Pflanzner, Z. Feher, and A. Kertesz, "A crawling approach to facilitate open iot data archiving and reuse," in 2019 Sixth International Conference on Internet of Things: Systems, Management

and Security (IOTSMS), 2019: IEEE, pp. 235-242.

[39] W. Zhu, Y. Li, Y. Xu, and X. Cui, "Optimal bandwidth allocation for web crawler systems," in 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), 2019: IEEE, pp. 1146-1153.

[40] A. Prusty, O. Mejia, A. Shah, P. Kancherlapalli, A. Suresh, and R. Schiebel, "Horizontally Scalable Web Crawler using Containerization and a Graphical user Interface."

[41] C. Pushpa, G. Deepak, A. Kumar, J. Thriveni, and K. Venugopal, "OntoDisco: improving web service discovery by hybridization of ontology focused concept clustering and interface semantics," in 2020 IEEE international conference on electronics, computing and communication technologies (CONECCT), 2020: IEEE, pp. 1-5.

[42] M. Biagiola, A. Stocco, F. Ricca, and P. Tonella, "Dependency-aware web test generation," in 2020 IEEE 13th International Conference on Software Testing, Validation and Verification (ICST), 2020: IEEE, pp. 175-185.

[43] U. Upadhyay, R. Busa-Fekete, W. Kotlowski, D. Pal, and B. Szorenyi, "Learning to crawl," in Proceedings of the AAAI Conference on Artificial Intelligence, 2020, vol. 34, no. 04, pp. 6046-6053.

[44] D. Arillotta et al., "Novel opioids: systematic web crawling within the e-psychonauts' scenario," Frontiers in neuroscience, vol. 14, p. 149, 2020.

[45] K. Avrachenkov, K. Patil, and G. Thoppe, "Online algorithms for estimating change rates of web pages," Performance Evaluation, vol. 153, p. 102261, 2022.

[46] M. Jakubíček, V. Kovář, P. Rychlý, and V. Suchomel, "Current challenges in web corpus building," in Proceedings of the 12th Web as Corpus Workshop, 2020, pp. 1-4.

[47] K. Jacksi and S. M. Abass, "Development history of the world wide web," Int. J. Sci. Technol. Res, vol. 8, no. 9, pp. 75-79, 2019.

[48] P. Hegade, R. Shilpa, P. Aigal, S. Pai, and P. Shejekar, "Crawler by inference," in 2020 Indo–Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN), 2020: IEEE, pp. 108-112.

[49] E. Uzun, "A novel web scraping approach using the additional information obtained from web pages," IEEE Access, vol. 8, pp. 61726-61740, 2020.

[50] X. Wang, Q. Sun, and J. Liang, "Json-ld based web api semantic annotation considering distributed knowledge," IEEE access, vol. 8, pp. 197203-197221, 2020.

[51] J. Xu, L. Du, C. Duan, and M. Li, "The application of web crawler in city image research," in 2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA), 2020: IEEE, pp. 270-274.