A Survey on Types of Crawlers and Web Searching Algorithms

T. Yogameera[1], Dr. D. Shanthi[2]

[1]Senior Lecturer, Department of Computer Engineering, Theni Kammavar Sangam Polytechnic College, Theni, Tamil Nadu, India
E-mail: *yogmeeraarasu[at]gmail.com*

[2]Professor and Head, Department of Computer Science and Engineering, PSNA College of Engineering and Technology, Dindigul, Tamil Nadu
E-mail: *dshan71[at]gmail.com*

Abstract: ***Information Analysis and gathering is one of the trending constrain in web data processing all the year around with increase in internet data and its dimensionality. Various fields like biometric, business analytics, security application and medical science are in need of the processed web information. It involves highly complex crawling and data searching techniques. The time and space complexity acts as the metric measure to justify the algorithm chosen for crawling, processing the needed information and retrieving it within the users contour. Traditionally search begins only by analyzing the Entry URLs from query submitted. Hence glimpse of matched links returned in buffer stacks fill in short time but the most relevant websites with real demanded data finds back in view. Hence the end user pursuit with many unnecessary links and continues inquest without satisfaction. This leads to in-efficient knowledge engineering and web resource utilization. This journal presents a survey on different search algorithms and techniques; also it discusses the types of web crawlers implementing the above search.***

Keywords: types of crawlers, searching algorithms used in crawlers

1. Introduction

A crawler is like a librarian. It looks for information on the Web, which it assigns to certain categories, and then indexes and catalogues it so that the crawled information is retrievable and evaluated.

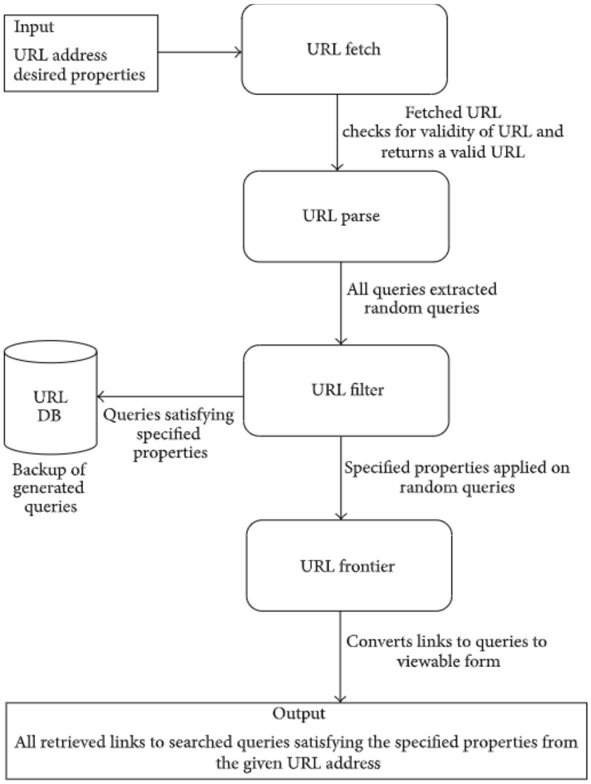
The crawler executes these instructions automatically. An index is created with the results of the crawler, which can be accessed through output software.The information a crawler will gather from the Web depends on the particular query.

The classic goal of a crawler is to create an index. Thus crawlers are the basis for the work of search engines. They first scour the Web for content and then make the results available to users. Focused crawlers, for example, focus on current, content-relevant websites when indexing.

Search engines have three primary functions:

1) **Crawl:** Scour the Internet for content, looking over the code/content for each URL they find.
2) **Index:** Store and organize the content found during the crawling process. Once a page is in the index, it can be displayed as a result to relevant queries.
3) **Rank:** retrieve the user needed best answer for a searcher's query, which means that results are ordered by most relevant to least relevant.

Block diagram of web crawler



Types of crawlers:

a) Google Bot

Googlebot is the most popular web crawlers on the internet today as it is used to index content for Google's search engine.

One great thing about Google's web crawler is that they give us a lot of tools and control over the process.

b) Bingbot
Bingbot is a web crawler deployed by Microsoft in 2010 to supply information to their Bing search engine. This is the replacement of what used to be the MSN bot. Bing also has a very similar tool as Google, called Fetch as Bingbot, .Request a page be crawled and shown to you as our crawler would see it. You will see the page code as Bingbot would see it, helping you to understand if they are seeing your page as you intended.

c) Slurp Bot
Collects content from partner sites for inclusion within sites like Yahoo News, Yahoo Finance and Yahoo Sports.Accesses pages from sites across the Web to confirm accuracy and improve Yahoo's personalized content for our users

d) DuckDuckBot
DuckDuckBot is the Web crawler for DuckDuckGo, a search engine that has become quite popular lately as it is known for privacy and not tracking you. It now handles over 12 million queries per day. DuckDuckGo gets its results from over four hundred sources. These include hundreds of vertical sources delivering niche Instant Answers, DuckDuckBot (their crawler) and crowd-sourced sites (Wikipedia). They also have more traditional links in the search results, which they source from Yahoo!, Yandex and Bing.

e) Facebook external hit
Facebook allows its users to send links to interesting web content to other Facebook users. Part of how this works on the Facebook system involves the temporary display of certain images or details related to the web content, such as the title of the webpage or the embed tag of a video. The Facebook system retrieves this information only after a user provides a link.One of their main crawling bots is Facebot, which is designed to help improve advertising performance.

f) Alexa crawler
It is the web crawler for Amazon's Alexa internet rankings. As you probably know they collect information to show rankings for both local and international sites

Context Graph Focused Crawler uses the limited capability of traditional search engines to allow users to query for pages linking to a specified document. This data can be represented as a graph which connected with each other and have a certain minimum distance necessary to move from one graph to another. This kind of data representation is used to train a set of classifiers to detect and assign documents to different categories depending on the expected link distance from the document to the target document. During the crawling process these classifiers are used to define the distance between a target document and a current processing document. And then all information is used to optimize and categorize the search.

There are two main stages of the algorithm:

a) An initialization phase when a set of context graphs and associated classifiers are constructed for each of the seed document;
b) A crawling phase that uses the classifier to lead the search process and updates context graph.

InfoSpiders are a multiagent system in which each agent in a population of peers adapts to its local information environment by learning to estimate the value of hyperlinks, while the population as a whole attempts to cover all promising areas through selective reproduction. The agent interacts with the information environment that consists of the actual networked collection (the Web) plus information kept on local data structures. It’s a type of agent-based systems that can browse online on behalf of the user, and evolve an intelligent behavior that exploits the Web’s linkage and textual cues. InfoSpiders is the name of online multiagent that search only the current environment and therefore will not return stale information, and will have a better chance to improve the regency of the visited documents. These agents dynamically crawl the Web in online mode, and uses artificial intelligence techniques to adapt to the characteristics of its networked information environment

Searching Algorithms used in crawlers:
Searching Algorithms are designed to check for an element or retrieve an element from any data structure where it is stored. Based on the type of search operation, these algorithms are generally classified into two categories:

1) The Fish Search Algorithm is an algorithm that was created for efficient focused web crawler. This system was implemented as a client based searching system tool that automatically navigates which webpage to crawl, thereby working more like a browsing user, but acting much faster and follows an optimized strategy. Client-based crawling have some significant disadvantages, like slow operation and resource consumption of the network.

Three types of actions:

1) The most important and difficult is the first step, it requires finding starting URLs, which will be the starting point of searching process.
2) Web documents are extracted and scanned for the information which is relevant at the receiving end.
3) The extracted web documents are reviewed to find links to other web documents or URL’s of webpages

Limitations of the search:

1) Crawling and downloading web documents through the WWW may be significantly time consuming, which is unacceptable for users.
2) The usage of resources of network is occasionally considered terrifically high. Compared to ordinary users visiting webpages and reading documents the fish search

crawlers significantly loads not only network, also web servers.

3) The algorithm can only retrieve documents for which URL's are found in other web pages. It means that this algorithm can't search documents from the "hidden" web.

2) Shark Search Algorithm is an improved version of the Fish Search algorithm. While this algorithm uses the same simple Fish School metaphor, it discovers and retrieves more relevant information in the same exploration time with improved search abilities. The Shark Search system uses better relevance scoring techniques for neighboring pages before accessing and analyzing them. The system have a significant impact on search efficiency because of improvement of the relevance classification system. It uses a score between 0 and 1, instead of the binary evaluation of information relevance. This approach gives much better results than binary classification. The second improvement is the method of inheritance of node's children. System gives every child an inherited score that have a huge impact on the relevance score of the children and children's children. And most significant improvement is that system calculates children's relevance using not only ancestor's heritage, but also use meta-data to analyze its relevance score. According to an experiment results the Shark Search is more effective in quality of information retrieved and operation time than its ancestor.

3) The Best-First algorithm focuses on the retrieval of pages which are relevant to a particular given topic. It's an algorithm that uses a score to define which page has a best score. This algorithm uses a rule to select the best page. In most cases it uses artificial intelligence algorithms (Naïve Bayes, Cosine Similarity, Support Vector Machine, k-nearest neighbor algorithm, Gaussian mixture model, etc.) as a classifier to detect the best result. In many articles this algorithm has the best crawling results. This algorithm is an algorithm of focused search which explores a graph by expanding the most hopeful node, this node selects according to a specific rule. The Best-first search algorithm principle is in evaluating the promise of node n by a heuristic technique estimation function f(n) which, generally, may depend on the specification of n, the specification of the goal, the information assembled by the search up to that point, and the most important, on any additional knowledge about the problem domain.

4) Learning Anchor Algorithm to define a relevance score of each link. This algorithm uses a classifier that assigns each link a different score depending on their anchor text information. This classification technique is more effective than the classification technique of the whole page. Also they use a method that combines the anchor and the whole parent document (using a Bayesian representation). Concluding experiment results we can conclude that anchor text alone can give much better information about the page pointed by the link than the document containing the link itself. But the main problem of this method is that anchor score does not contribute to the evaluation when the anchor text doesn't contain any information. The figure 6 presents a pseudocode of the algorithm This approach helps a focused crawler to assign better priorities to the unvisited links in the crawl frontier that leads to a higher rate of fetching pages and decrease false events that positively influence on PCU, network and storage resources.

5) The Context Graph Focused Crawling Algorithm uses the limited capability of traditional search engines to allow users to query for pages linking to a specified document. This data can be represented as a graph which connected with each other and have a certain minimum distance necessary to move from one graph to another. This kind of data representation is used to train a set of classifiers to detect and assign documents to different categories depending on the expected link distance from the document to the target document. During the crawling process these classifiers are used to define the distance between a target document and a current processing document. And then all information is used to optimize and categorize the search. There are two main stages of the algorithm: an initialization phase when a set of context graphs and associated classifiers are constructed for each of the seed document; and a crawling phase that uses the classifier to lead the search process and updates context graph.

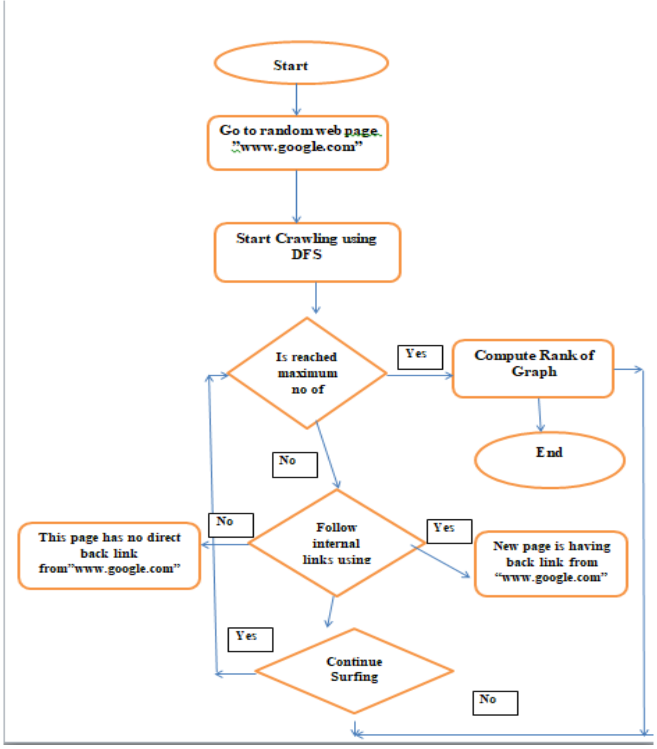
6) Depth First Algorithm, For applications of DFS in relation to specific domains, such as searching for solutions in artificial intelligence or web-crawling, the graph to be traversed is often either too large to visit in its entirety or infinite (DFS may suffer from non-termination). In such cases, search is only performed to a limited depth; due to limited resources, such as memory or disk space, one typically does not use data structures to keep track of the set of all previously visited vertices. When search is performed to a limited depth, the time is still linear in terms of the number of expanded vertices and edges (although this number is not the same as the size of the entire graph because some vertices may be searched more than once and others not at all) but the space complexity of this variant of DFS is only proportional to the depth limit, and as a result, is much smaller than the space needed for searching to the same depth using breadth-first search. For such applications, DFS also lends itself much better to heuristic methods for choosing a likely-looking branch. When an appropriate depth limit is not known a priori, iterative deepening depth-first search applies DFS repeatedly with a sequence of increasing limits. In the artificial intelligence mode of analysis, with a branching factor greater than one, iterative deepening increases the running time by only a constant factor over the case in which the correct depth limit is known due to the geometric growth of the number of nodes per level.

Input: A graph *G* and a vertex *v* of G

Output: All vertices reachable *S*.push(*v*)

```
procedure DFS(G, v) is
label v as discovered
for all directed edges from v to w that are in G.adjacentEdges(v) do
if vertex w is not labeled as discovered then
recursively call DFS(G, w)
```

```
procedureDFS_iterative(G, v) is
letS be a stack
S.push(v)
while S is not empty do
 v = S.pop()
if v is not labeled as discovered then
label v as discovered
for all edges from v to w in G.
adjacentEdges(v) do
S.push(w)
```


Start
Go to random web page "www.google.com"
Start Crawling using DFS
Is reached maximum no of
Yes
Compute Rank of Graph
End
No
No
Follow internal links usine
Yes
This page has no direct back link from"www.google.com"
New page is having back link from "www.google.com"
Yes
Continue Surfing
No

Comparison tables

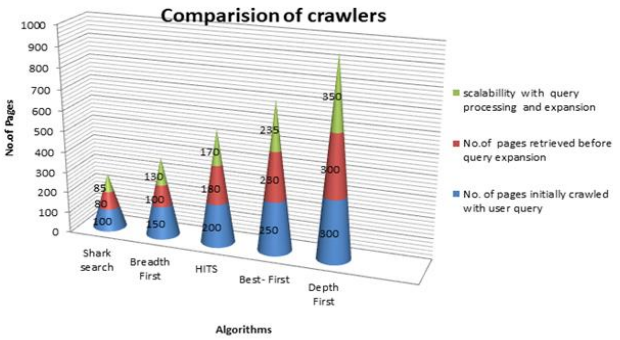
Table 1-URLs with most links

	# of pages visited	# of relevant pages visited
Breadth First	100	28.79
Best First	100	33.93
Page rank	100	35.09
Shark Search	100	27.06
HITS	100	31.23
Depth first Search	100	59.32

Table 2 -URLs with Topic Score

	# of pages visited	# of relevant pages visited
Breadth First	100	32.52
Best First	100	34.26
Page rank	100	33.63
Shark Search	100	31
HITS	100	32.6
Depth first Search	100	40

Fig1: Comparison chart for analysis:



2. Conclusion

The choice of the algorithm has a significant impact on the work and effectiveness of focused crawler and search engine. According to research the best crawling algorithm is the Best First search algorithm using artificial intelligence classifiers like Support Vector Machine, Naïve Bayes Classifier, String Matching, etc. Some researchers use semi-supervised learning methods to crawl and analyze the crawled information .The best method to crawl most relevant information is to use unsupervised learning methods combined with focused crawling algorithms that defines the best result of currently crawling pages. Web Search Engines face new challenges due to the availability of vast amounts of web documents, thus making the retrieved results less applicable to the analyzers. However, recently, Web Crawling solely focuses on obtaining the links of the corresponding documents. Today, there exist various algorithms and software which are used to crawl links from the web which has to be further processed for future use, thereby increasing the overload of the analyzer. Further concentration is needed for crawling the links and retrieving all information associated with them to facilitate easy processing for other uses. To conclude firstly the links are crawled from the specified uniform resource locator (URL) using a modified version of Depth First Search Algorithm which allows for complete hierarchical scanning of corresponding web links. The links are then accessed via the source code and its metadata such as title, keywords, and description are extracted. This content is very essential for any type of analyser work to be carried on the Big Data obtained as a result of Web Crawling

References

[1] Diana Inkpen, "Information Retrieval on the Internet", Assistant Professor, University of Ottawa, Canada, KIN6N5. (journal style)

[2] Chris Manning, Pandu Nayak, and Prabhakar Raghavan, Information Retrieval and Web Search, Computer Science Department, Stanford University, Stanford, CA 94305, USA. (journal style)

[3] Y. Kalmukov and I. Valova, "Design and development of an automated web crawler used for building image databases, " 2019 42nd International Convention on Information and Communication Technology, Electronics

and Microelectronics (MIPRO), Opatija, Croatia, 2019, pp. 1553-1558, doi: 10.23919/MIPRO.2019.8756790.

[4] Web crawler analysis sites like dynomapper, Oncrawl, Oxylab,

[5] G. H. Agre and N. V. Mahajan, "Keyword focused web crawler, " 2015 2nd International Conference on Electronics and Communication Systems (ICECS), Coimbatore, 2015, pp. 1089-1092, doi: 10.1109/ECS.2015.7124749.

[6] S. GOEL, M. BANSAL, A. K. SRIVASTAVA and N. ARORA, "Web Crawling-based Search Engine using Python, " 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2019, pp.436-438, doi: 0.1109/ICECA.2019.8821866.

[7] C. Saini and V. Arora, "Information retrieval in web crawling: A survey, " 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Jaipur, pp.2635-2643, doi: 10.1109/ICACCI.2016.7732456.

[8] N. Ragavan, Y. Rubavathi C. and J. Singh K., "Crawler Framework for Category Search Engine, " 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), Vellore, India, 2020, pp. 1-5, doi: 10.1109/ic-ETITE47903.2020.167.