**Internet and Web Systems I (Project)**

**Week 2**

**By Surbhi Kanthed and Ankit Nimje**

**WEB CRAWLER**

**Keywords**:

Frontier, web indexing,

What is a Web Crawler?

According to *Wikipedia*, “A Web crawler, sometimes called a spider or spiderbot and often shortened to crawler, is an Internet bot that systematically browses the World Wide Web, typically for the purpose of Web indexing (web spidering).”

Search Engines such as Google and some web sites use Web Crawlers to update their web contents. Web crawlers make indices of visited web pages thus search engines can process more efficiently.

How does Google search works?

In the world wide web, there are more than 6 trillion individual web pages. Google keeps track of these webpages by crawling through them and makes index of these web pages. When an user gives a search query, google engine matches it with its indexed terms using generated algorithms while taking care of number of features such as - spelling, search methods, synonyms, autocomplete, Google instant, etc.

A crawler starts exploring from a seed page and covers all the hyperlinks within current page. This process is repeated until sufficient number of pages are crawled or some objective is achieved. As we know web is a dynamic entity with content being updated at rapid rates. Were the web being a static collection of pages, we would have fetched all the data and stored in system repository to use it for longer duration.

So far as now, we have studied the flow of basic sequential crawler as shown in Figure 1. As we start, the crawler initializes with list of seed URL’s called as frontier. Then starts the crawling loop. It starts with picking the topmost URL from the frontier then fetches it. After parsing it to look for all the hyperlinks in that web page, crawler adds those links into frontier seed list to be fetched. Then a it checks for a termination condition. If the termination condition fulfills it stops or continues crawling cycle. Crawling can be thought as a graph search problem. The Web can be viewed as a large graph with pages as its nodes and hyperlinks a site edges. A crawler starts at a few of the nodes and then follows the edges to reach other nodes [2].

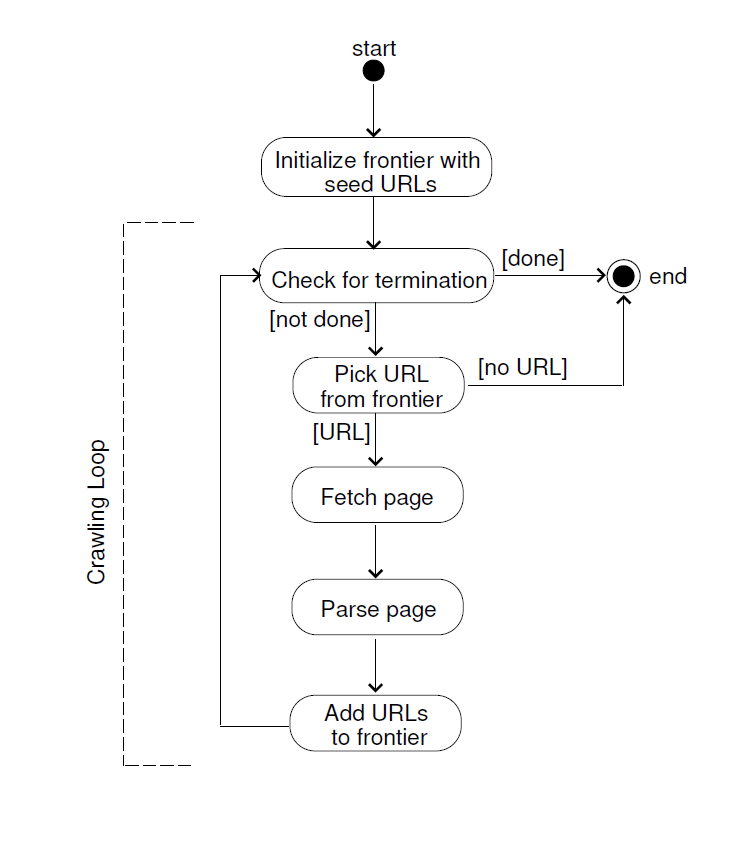


Figure 1: Flow of Basic Sequential Crawler

**Implementation:**

Web crawler seems to be an uphill battle, so we just started with crawling the page and try to get the HYPERLINKS associated to it. We studied different methods to perform tasks like:

* Creating the directory of the website
* Creating different files to store links crawled and queued
* Adding and deleting queued links from file if crawled and add to another file.

Also, we converted byte code of website crawled into HTML code using HTML parser and defined method to find all the hyperlink of webpage and place it in our queue file.

**Challenges Faced:**

In most cases, when pages are crawled it contains links associated with Facebook, Twitter, Gmail, YouTube if those links are added in queue it again has near-infinite links associated with it, in that case crawler goes in infinite loop.

To overcome this, we have defined method to identify the domain or homepage of the website. We made sure that links which are being crawled are related to current website.

References:

1] Medium - Everything you need to know about a Web Crawler

https://medium.com/@cabot\_solutions/web-crawlers-everything-you-need-to-know-6dce26ee8ad8

2] Gautam Pant, Padmini Srinivasan, Filippo Menczer - ‘Crawling the Web’

3] Bucky Roberts – ‘TheNewBoston’