Web Search: Web Crawling

COMP3009J: Information Retrieval

Dr. David Lillis (david.lillis@ucd.ie)

UCD School of Computer Science
Beijing Dublin International College

Introduction

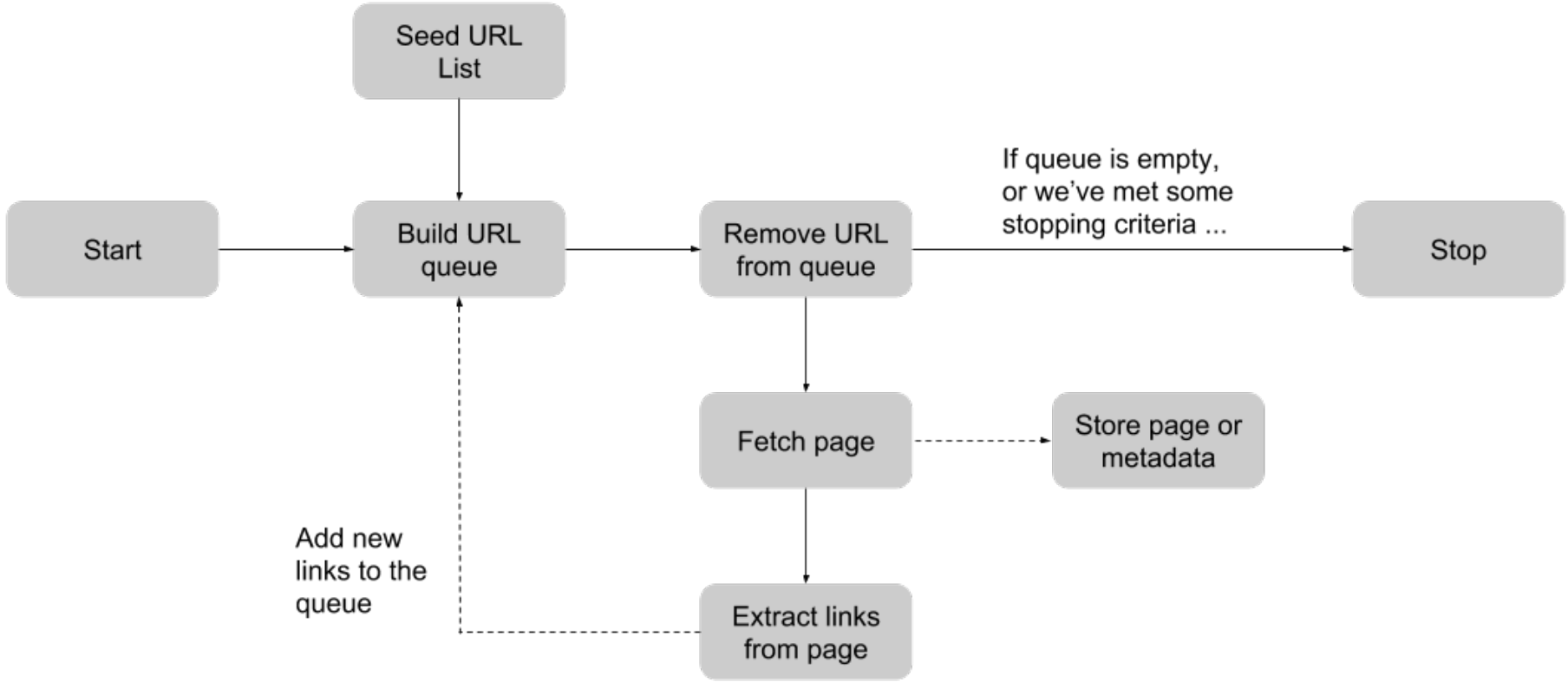
- Many of the techniques we have look at, both for performing and evaluating Information Retrieval were originally developed before the World Wide Web (WWW) became popular.
- Nowadays, the WWW is the biggest source of information in existence and web search has become an extremely important aspect of using the web.
- With the shift of IR towards this new environment, a number of significant challenges have presented themselves. We will look at a number of these.

How do we find information?

- **Web Crawlers** or **Web Spiders** are programs that automatically look for documents on the web that will be included in the index so that users can find them when searching.
- They do this by **extracting the hyperlinks** from each page they find, which will in turn point them to new pages to index.
- Even the decision on what policy to use for following hyperlinks is a subject of debate: one study claims that using a Breadth-First Search strategy will result in finding higher-quality pages (as measured by PageRank)*.
- It is also considered to be more polite to use a Breadth-First Search, as it reduces load on servers.

* M. Najork and J. Weiner, Breadth-First Search Crawling Yields High-Quality Pages, 2000

Typical Web Crawler Workflow



Source: https://medium.com/@allisonmorgan/short-essay-on-web-crawling-scraping-8abf1b232b65

How do we find information?

- Breadth-first spider search:
  - Manually add a list of "seed" URLs to the spider's queue of pages to visit
  - While there are still URLs in the queue of pages to visit
    - Remove the first URL from the queue of pages to visit and download the page
    - Extract the links from the document
    - Add these links to the queue of pages to visit if they haven't already been visited and are not already in the queue.
- A depth-first search is the same, except you use a stack instead of a queue.
- Normally runs in **parallel**, so it does not waste time waiting for websites to respond.

Polite Crawlers

- It is easy to harm a website by using a web crawler, especially by overloading it with too many requests.
- Web Crawlers are expected to be “polite”, by:
  - Obeying robots.txt files and other methods to give instructions to crawlers.
  - Not overloading the website (obey Crawl-Delay).
  - Not consuming too much bandwidth.
  - Identifying who they are and who they belong to (“Googlebot”, “Baiduspider”).

Polite Crawlers: The Robots Exclusion Standard

- The Robots Exclusion Standard is a method of asking web crawlers not to index certain portions of a web site.
- It is a **voluntary standard**, so there are no technical barriers stopping crawlers that ignore the standard from accessing the pages in question.
- Whenever a well-behaved crawler visits a web server, it will firstly check the root directory for a file called `robots.txt` (i.e. `http://www.example.com/robots.txt`).
- This is a plain text file that contains a list of resources that the crawler should not visit.
- Different lists can be provided for different crawlers (e.g. if you wanted a particular page to be searchable on Yahoo! but not on Google).

Finding Information - the Robots Exclusion Standard

- Robots Exclusion Rules

```
User-agent: *
Disallow: /cgi-bin/
Disallow: /tmp/
Disallow: /~joe/
```

- Allow one bot

```
User-agent: Googlebot
Disallow:

User-agent: *
Disallow: /
```

robots.txt

- There are a number of common mistakes made by web site owners when creating robots.txt files:
  - The robots.txt file must be in the root directory. A crawler is not required to check anywhere else.
  - Only one disallowed path should be specified on each line.
  - Missing forward-slash at the beginning of disallowed paths.
  - There are no wildcards in the "Disallow" section (though some crawlers support it)
  - Misspelling the name of a crawler you wish to exclude.
  - Trying to use the non-existent "Allow" command (though some crawlers support it)
- Some crawlers obey a "Crawl-Delay" instruction (how long to wait before the next request).

Polite Crawlers: noindex and nofollow

- Ignore pages that have `<meta name="robots" content="noindex" />`.
- Don't follow any links from pages that have `<meta name="robots" content="nofollow" />`.
- Individual links can be marked `nofollow` also:

```
<a href="http://example.com">Anchor Text</a>
<a href="http://example.com" rel="nofollow">Anchor Text</a>
```

- The exact treatment of this varies for different search engines.

Non-static documents

- Another issue is the fact that the web is a **dynamic environment.**
- New pages are **added constantly,** and many existing pages may **change frequently**.
- Even if a crawler has visited a large number of documents, it must still be **revisited** so the index does not get out-of-date.
  - For instance, news and social media sites will be constantly changing and are typically re-indexed on a regular basis.
- On the other hand, there are huge numbers of pages that are essentially static over long periods of time (mailing list archives, individual news articles, etc.)
- A policy must be implemented that decides which pages should be re-indexed on a regular basis and which need not be.

Finding information: challenges

Pages designed to frustrate attempts at automated access (e.g. through the use of CAPTCHAs).

A CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) is typically an image containing text that computers find difficult to read.

They are frequently to be seen when signing up for accounts with various web sites.

An automatic crawler can't access pages that are guarded by CAPTCHAs.

How big is the web?

- Google state that their index contains “hundreds of billions of webpages”.
- Technically, the size of the web is probably infinite, as dynamic pages give different results depending on the information they’re given:
  - Calendar applications can typically show data for an indefinite amount of time into the future.
  - Search engines themselves will give a different page in response to every different query they receive.

* http://www.google.com/insidesearch/howsearchworks/thestory/

How big is the web?

- The **Deep Web*** (or invisible web or hidden web) refers to the fact that web crawlers cannot easily gain access to particular online resources.
  - Some pages or groups of pages may not have any links from elsewhere. Since web crawlers work by following links, it may not be possible to find these pages.
- Content served via AJAX/Asynchronous JavaScript or Flash can be hard for crawlers to index. AJAX content is fetched by JavaScript code running within a web page and is not accessible just using a URL like traditional content.

* M. K. Bergman, The Deep Web: Surfacing Hidden Value 2001