Project - Web crawler and scraper - I

Existing similar repository of assigned project - https://github.com/amirgamil/apollo

A web crawler and search engine for analysing your digital footprint. This implies that you get to decide what goes inside. You manually add it when you find something that seems intriguing, whether it's an article, blog post, website, or anything else (with built in systems to make doing so easy). You can also choose to consistently import data from a certain data source, such as your notes or another source. As a result of Apollo's high signal-to-noise ratio, one of the main issues with search engines returning a lot of unnecessary information is addressed. You specifically decided what to put in there.

Local records and data from data sources are stored in separate JSON files.

Data comes in many forms and the more varied those forms are, the harder it's to write reliable software to deal with it. If everything I wanted to index was just stuff I wrote, life would be easy.

A web crawler is a program that automatically navigates the web by browsing links and downloading content from websites. The purpose of a web crawler can range from indexing the web for search engines to analysing websites for specific information.

A web scraper is a program that extracts specific information from websites. It is used to extract data from websites for a variety of purposes such as data analysis, data migration, or to gather specific information for a database. Web scraping can be done manually or with the use of a web scraper program.

Both web crawlers and web scrapers are important tools for data extraction and analysis, and they are widely used in various industries.

The web crawler should be flexible enough to allow users to define which types of pages they want to crawl and how they want the data to be structured.
Resilience: A web crawler should be able to handle unexpected situations, such as broken links or server errors, and continue crawling the rest of the website and the web scraper should be able to handle websites with dynamic content, such as pages that use JavaScript to load content.

- A ported version of the Go snowball algorithm is used for the stemmer. This might be making the algorithm for the search engine slower. There might be further improvements which might be helpful to make the search engine faster.
- In the given repository, only the admin can access the add data since there was authentication before adding the scraped data. We are planning to make it an open source scraper where people can add constrained data like e books, blogs, etc.
- The search engine is in Go but we will try implementing a more efficient crawler and crawler in Python.
- The search engine can be improved for scrapping.
- The crawler is having vivid uses so a further improvement can always be possible.