**Internship Coding Assignment**

**Summary**

**-Likith K**

The report presents a Python Script that was developed to gather information about Canoo, a publicly trading company. The task presented was to search the internet for relevant information pertaining to Canoo from various online sources, scraping the retrieved information from the links, and store it in a structured tabular format in a CSV file for further analysis.

Retrieval Augmented Generation is the process of optimizing the output of a large language model, so it references an authoritative knowledge base outside of its training data sources before generating a response. This process is quite useful as it prevents hallucinations as well as provides reliable information of generative AI models. The script utilizes internet search APIs to perform searches for specific queries related to Canoo. Upon obtaining search results, it proceeds to scrape data from the links returned by the Internet Search API. The scraped data is then organized into a structured tabular format and stored in a CSV file, ensuring easy accessibility and manipulation for subsequent analysis.

Steps involved:

There are essentially three phases to this process, the first one is the Internet Search API, the scraping phase, and the data storage phase:

1. Internet Search API: The queries are stored in a list need to be individually searched using an Internet Search API. The API chosen is the DuckDuckGo API, as it is easy to use while being free of cost as other Internet Search API comes with a trial version or a paid service. The DuckDuckGo API individually searches each query, and the links are saved in a new CSV file called search\_results, the title and the URL are stored in the CSV file.
2. Scraping Phase: The Scraping Python library used for the task is the BeautifulSoup library that parses the HTML of the URL. The biggest issue with using other Python libraries would be the different ways the data is stored in the webpages, extracting very specific information is difficult, hence the BeautifulSoup library is used to extract all the text that’s in the <p> tag as well as the <span> tag. Using XPath to extract is time consuming and complicated when the number of URL is large, and they have different class names. Another approach would be using the LangChain library where the query is sent to an Internet Search API and the retrieved URL is scraped using the BeautifulSoup library then send the data to a LLM that extracts data using the provided Schema. This method utilizes the OpenAI API which is a paid service.

So, by using the much simpler and basic BeautifulSoup library, text is scrapped and stored in a CSV file.

1. Data Storage: The file CSV is generated using the Pandas library by creating a data frame with the title, URL, and the text.

Challenges faced and Overcame:

1. The Internet Search API chosen was the DuckDuckGo search library since it is free. Other APIs such as Brave have a limit on the number of requests made to the API.
2. The HTML structure of each URL is different hence specific data cannot be extracted using the XPath or Class name methods. Hence the text from <p> tags were scrapped.
3. The time to scrap the URL was high, hence, to overcome this challenge Asynchronous Web Scraping was used using the AIOHTTP and asyncio module. Asynchronous web scraping is a special technique that allows you to begin a potentially lengthy task and still have a chance to respond to other events, rather than having to wait for that long task to finish.
4. Some URLS are not responsive, therefore did not have the 200-status code. To overcome this the status code of all the URLs was checked and only the URLs with the 200 status codes were scrapped.
5. The text scrapped from the URL were extremely large, therefore data was limited to 2000 words.
6. Some URL do not store text data in <p> tags or <span> tags hence these URL have empty text field.
7. Introducing a Language Model will highly benefit the process of summarizing the text data as well as reading the raw HTML file more efficiently and make it easier for the next step of the RAG process, but it comes with a price.

Libraries Used:

1.Pandas

2. BeautifulSoup

3. asyncio

4.aiohttp

5.csv

6. DuckDuckGo Search