MA5851 A3 Assessment Report Part Two (WebCrawler)

Student: Sacha Schwab

Location: Zurich, Switzerland

Date: 3 December 2021

Github link to repository: https://github.com/sachaschwab/NLP-Clustering (https://github.com/sachaschwab/NLP-Clustering)

Note:

Figures (images) are not correctly loaded by github file; please use the documents submitted on JCU Learn in the assessment submission folder

a) Websites consumed

Yahoo Finance (crawled with BeautifulSoup)

Yahoo Finance collects news articles from multiple sources such as Bloomberg and Reuters. The advantage is that the articles are presented in Yahoo's html format. Also, it features a "Cryptocurrency" news section. Therefore, and since I have not found or encountered any limitations to webcrawling, this resource appears appropriate for the task at hand.

URLs:

- Main site: https://finance.yahoo.com/topic/crypto/
- Individual articles may have different prefix urls, however the crawler is flexible in that regard, since the individual article URLs are extracted from the main site.

Yahoo Cryptocurrencies (crawled with Selenium)

As to be seen in the NLP part of this project, a list of current cryptocurrencies is required for injection into e.g. named entitiy extraction. Since the names are part of a table which takes on 25 items, the user needs to click through all the tables to get the names of all cryptocurrencies.
Therefore, Selenium was used to click the button element, getting the next 25 items, and so on, until all approx. 375 names are retrieved.

URL: https://finance.yahoo.com/cryptocurrencies

b) Rationale for extraction

The aim of this project requires not only the gathering of high-quality news text, but also to achieve a corpus of the size indicated in the assessment outline, i.e. a minimum of 100-300 documents, since the texts appear medium sized. Further, it is beneficial to collect article texts from various domains so to avoid uniformity of the format i.e. to produce a model that covers a wider range of text structure and lengths. Yahoo Finance ticks all these requirements and therefore appears to be a valid choice.

c) Content coverage

A preliminary manual review of Yahoo Finance (Cryptocurrencies) revealed that the number and range of articles appears interesting, since it features not only brief market event comments (such as with Bloomberg) but also developing stories. This covers the purpose of this project, i.e. to provide per-event-clustering of news articles.

d) Complexity of the content layout

The html content layout is not highly complex, however requires some html skills to de-code it for the purpose of webcrawling, in particular since the pages are rendered by react-js engine. Therefore, the tag classes are presented can be quite tricky when it comes to interpreting which substring actually triggers the tag, such as in 'caas-xray-wrapper caas-xray-wrapper-type-cards caas-xray-wrapper-position-top'. However, in the end the tags turn out to be quite straight forward.

e) Website/data copyright considerations

Permitted guidelines check:

- Public data only: Yahoo.com is public. There is no walled garden, and neither is it a paid service.
- Previously allowed: A large number of resources was found on scraping Yahoo.com content; it therefore appears that Yahoo implicitly allows webcrawling.
- Non-copyright-protected content: The content under the above mentioned URL does not contain any copyright protection notice and, at the time of issuing this report, it was not found under the CloudFlare protected website search.

f) Metadata supplementation

Supplementation of the articles' dates turned out to be sufficient, and the author does not appear relevant for the purpose of this project.

g) Content extractor / WebCrawler workflow

- The **technology component** used for the web crawler is limited to Beautifulsoup, which appeeared sufficient for the purpose of this project. There was no need to employ Selenium (e.g. to render older articles to the html content) since, to keep the data and the model up to date, a daily run of the crawler is anyway necessary.
- As outlined above, the **ccmplexity of the domains** is rather low. The **targeted data** resides in the article pages, URLs of which were obtained by extracting the hrefs from the main page.
- Some **sequencing** was applied by first processing the URLs of the new articles in the raw data frame csv, with subsequent crawling of the individual articles pages in a loop.
- **Data storage** is achieved using csv format, which in light of the limited size of the data and for performance considerations appears appropriate.

Basic workflow:

workflow

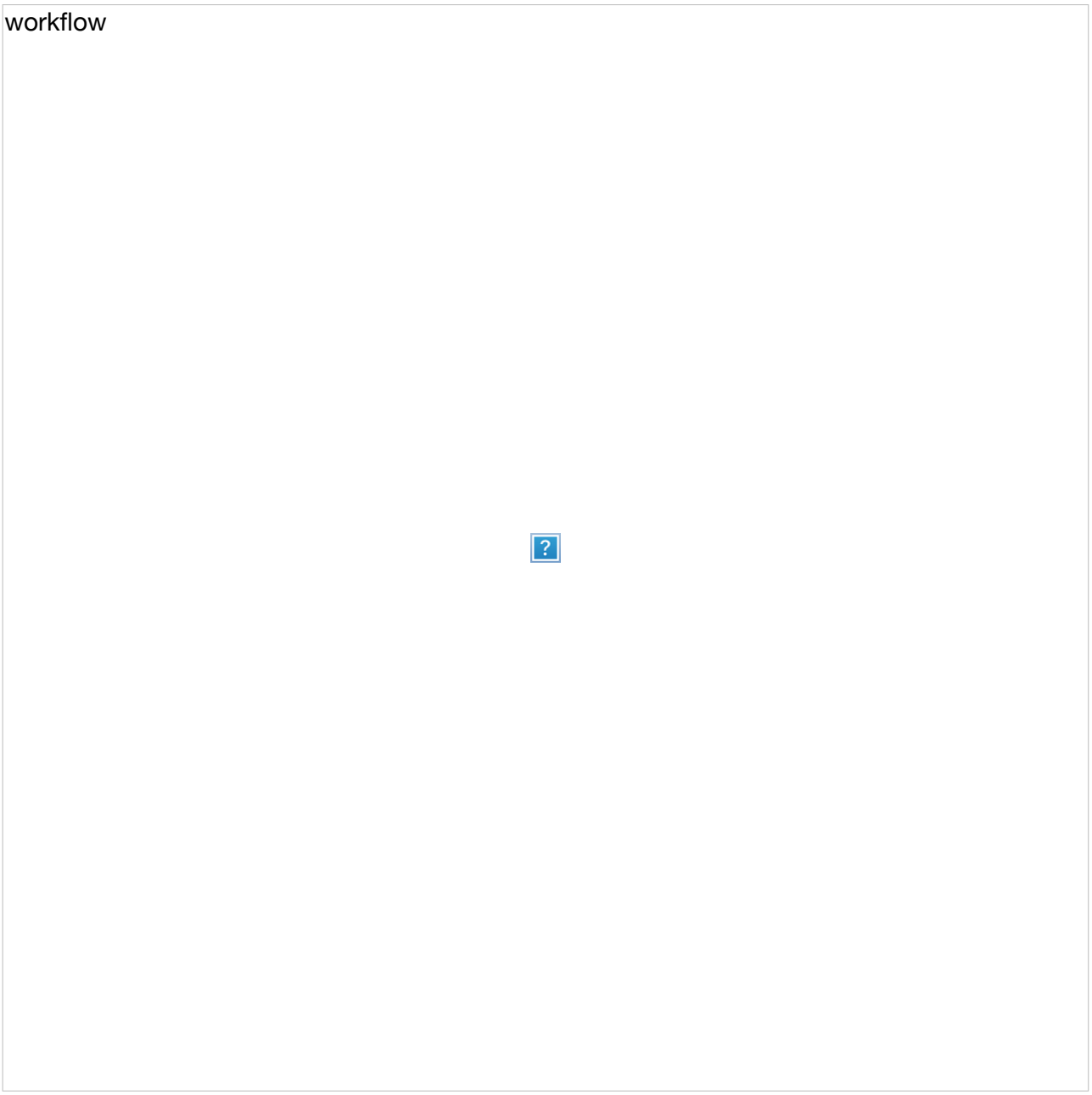

Figure: Webcrawler workflow

h) Python coding

Python code as per code files (see links below) use PEP8 and PEP256 code format.

Yahoo Finance (articles)

i) Demonstration of the application of the WebCrawler

Yahoo main page:

Figure: Yahoo main page with inspection of the title

An article page sample:

article page

Figure: Yahoo individual article page with inspection of the title

Daily webcrawling gif:

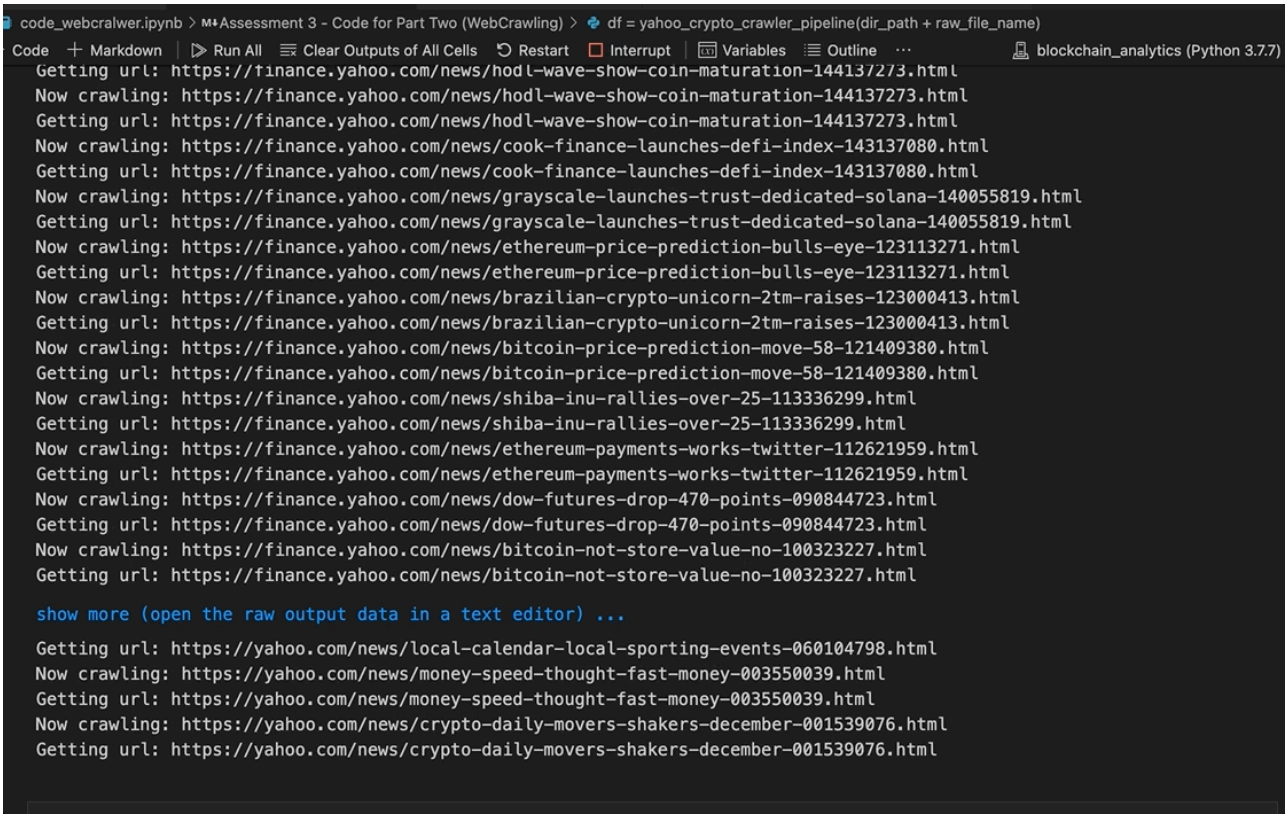

Figure: Running webcrawler (gif)

Yahoo Cryptocurrencies list

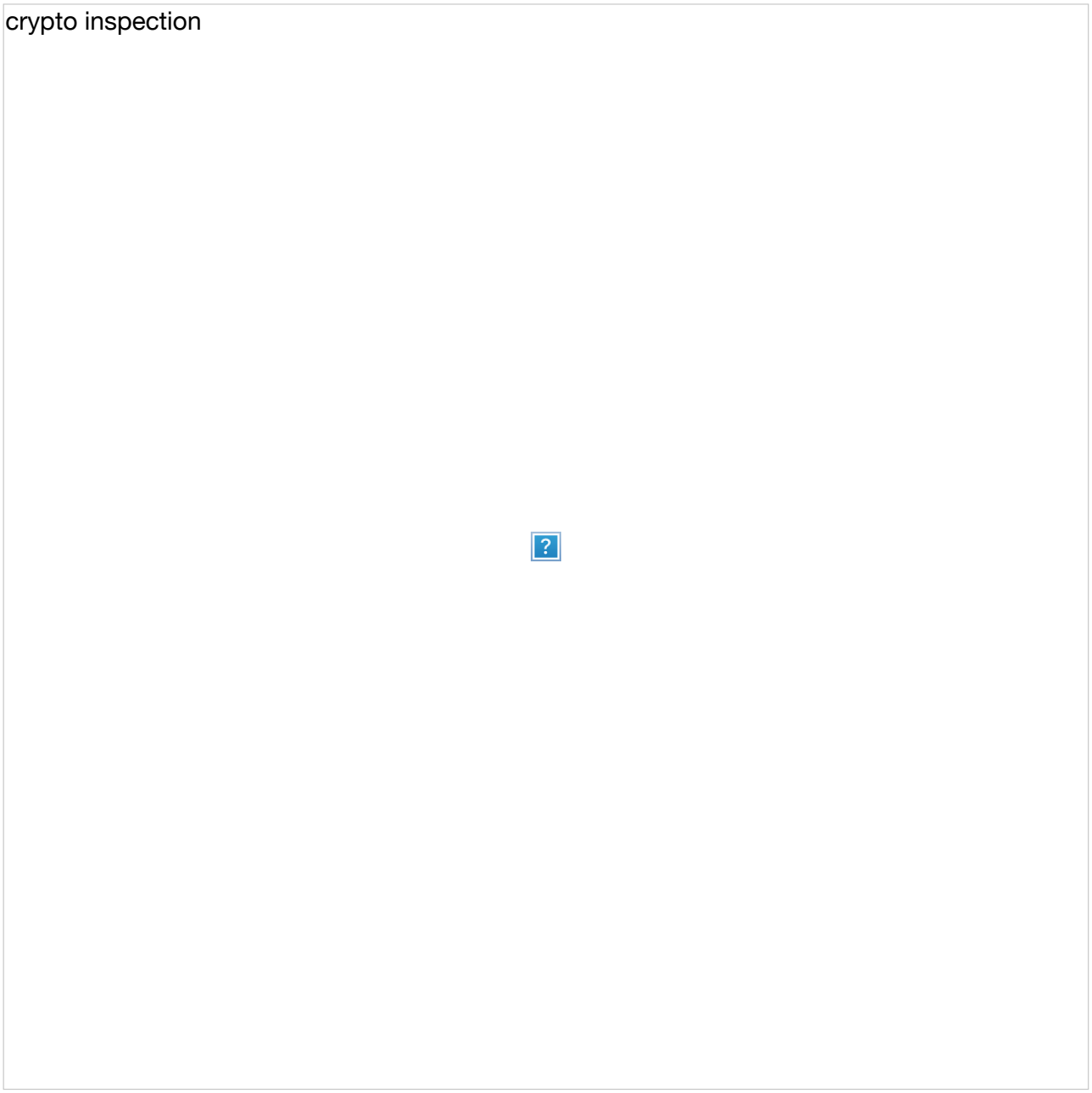

Figure: Yahoo Cryptocurrencies list - Inspection of button element to repeatedly click

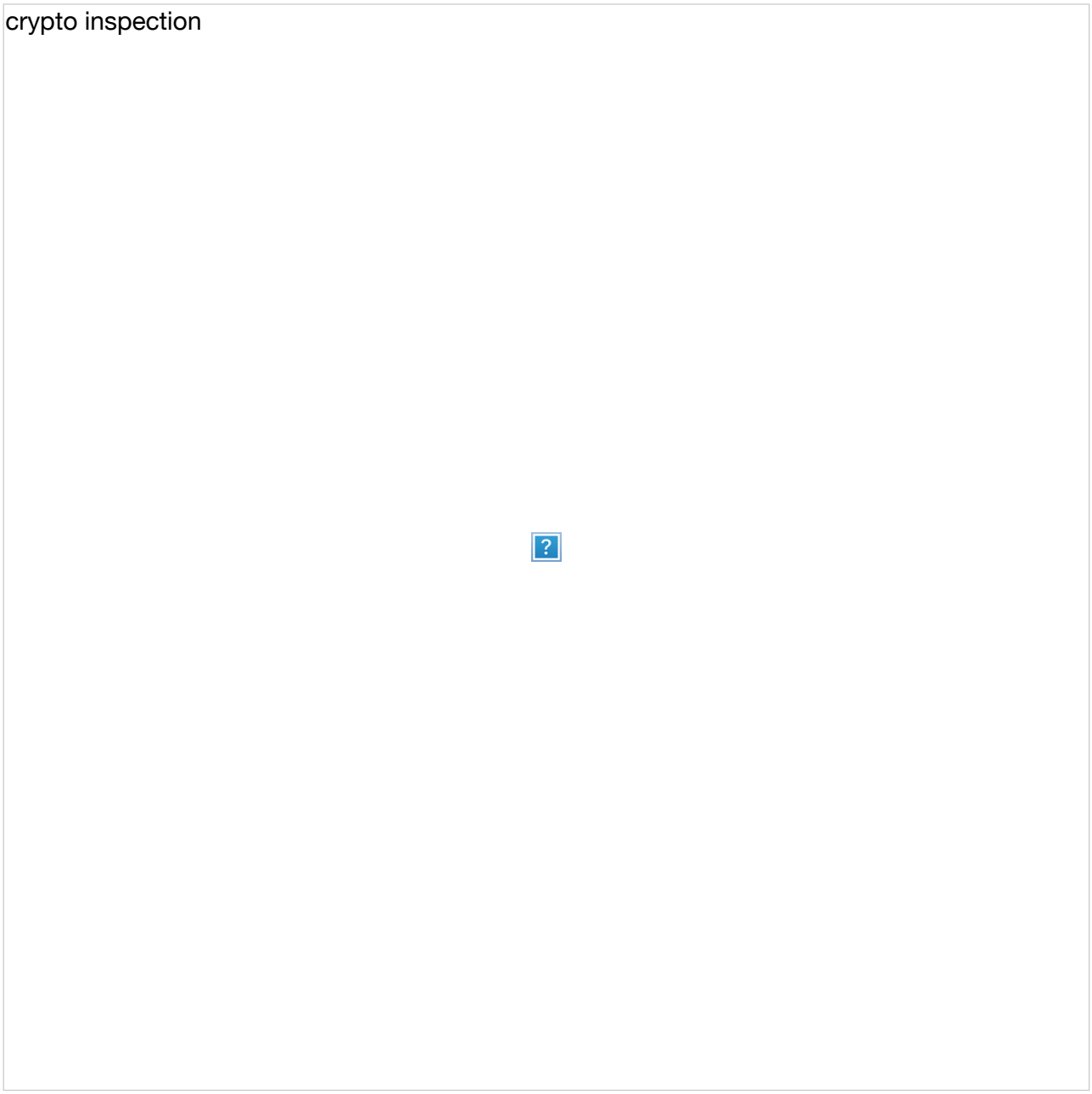

Figure: Gif of activity log while crawling cryptocurency names with Selenium

j) Methodology of processing, cleaning, and storing harvested data for NLP tasking

The raw text data is stored as csv file. For preparation of the NLP tasks, removal of symbols, stop words performed, and the text is lemmatizsed usinng the NLTK WordLemmatizer. Also, lower casing is applied. Reference is made to document 3 (NLP tasks report). This procedure appears appropriate in light of the many resources recommending these steps in particular for TF-IDF vectorisation. However, resource would usually apply stemming instead of lemmatisation, however the latter is a personal preference since the output makes more sense.

k) Summary and visualisation of the harvested data

The following summary reflects the status as per 5 December 2021

- Number of documents: 385
- Date range: From 2021-11-22 to: 2021-12-04
- Mean word length of article body texts: 286.32987012987013
- Total corpus size: 822565

Frequency distribution of words in titles and body texts:

article page

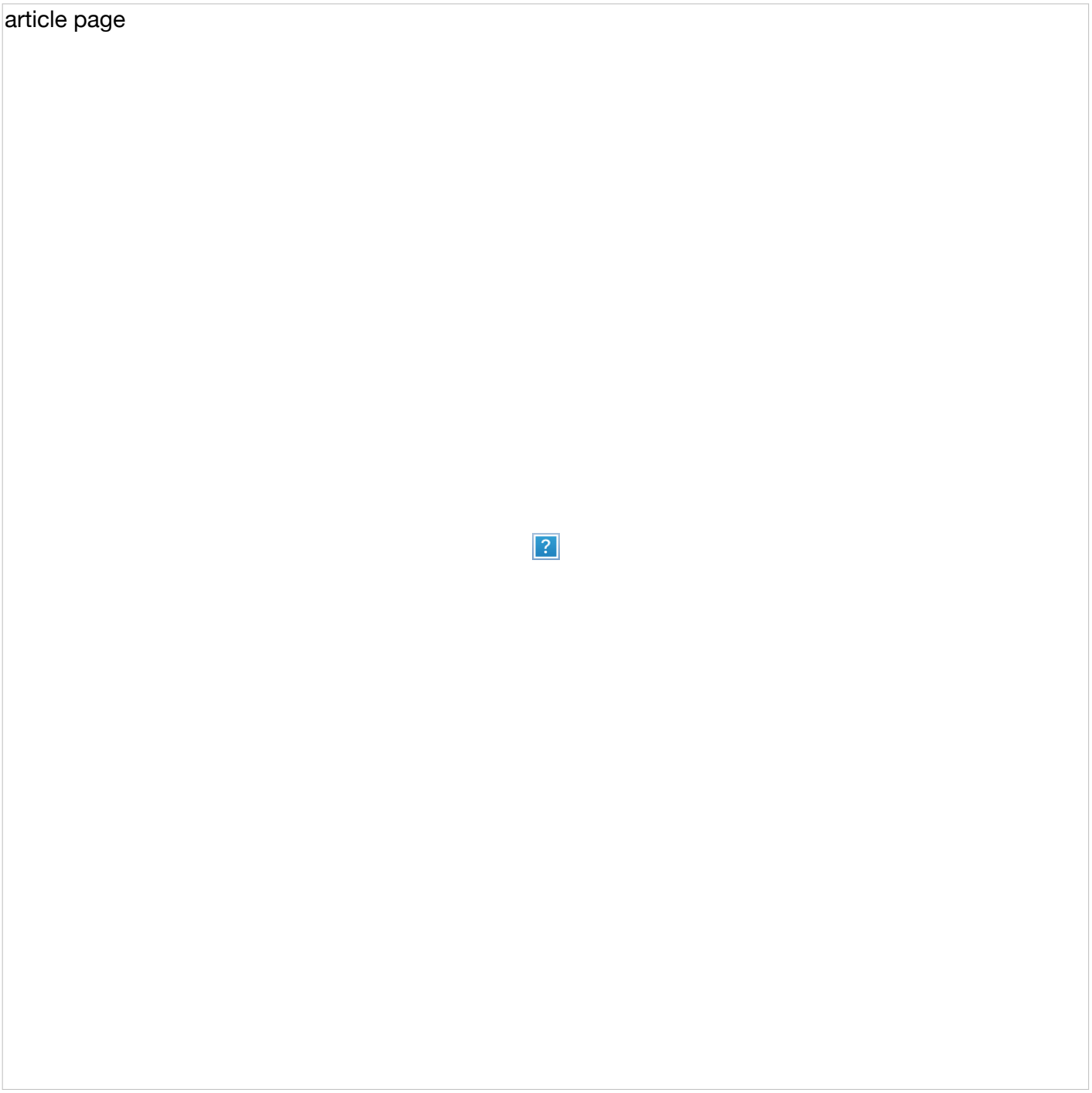

Figure: Distribution of word count in titles

article page

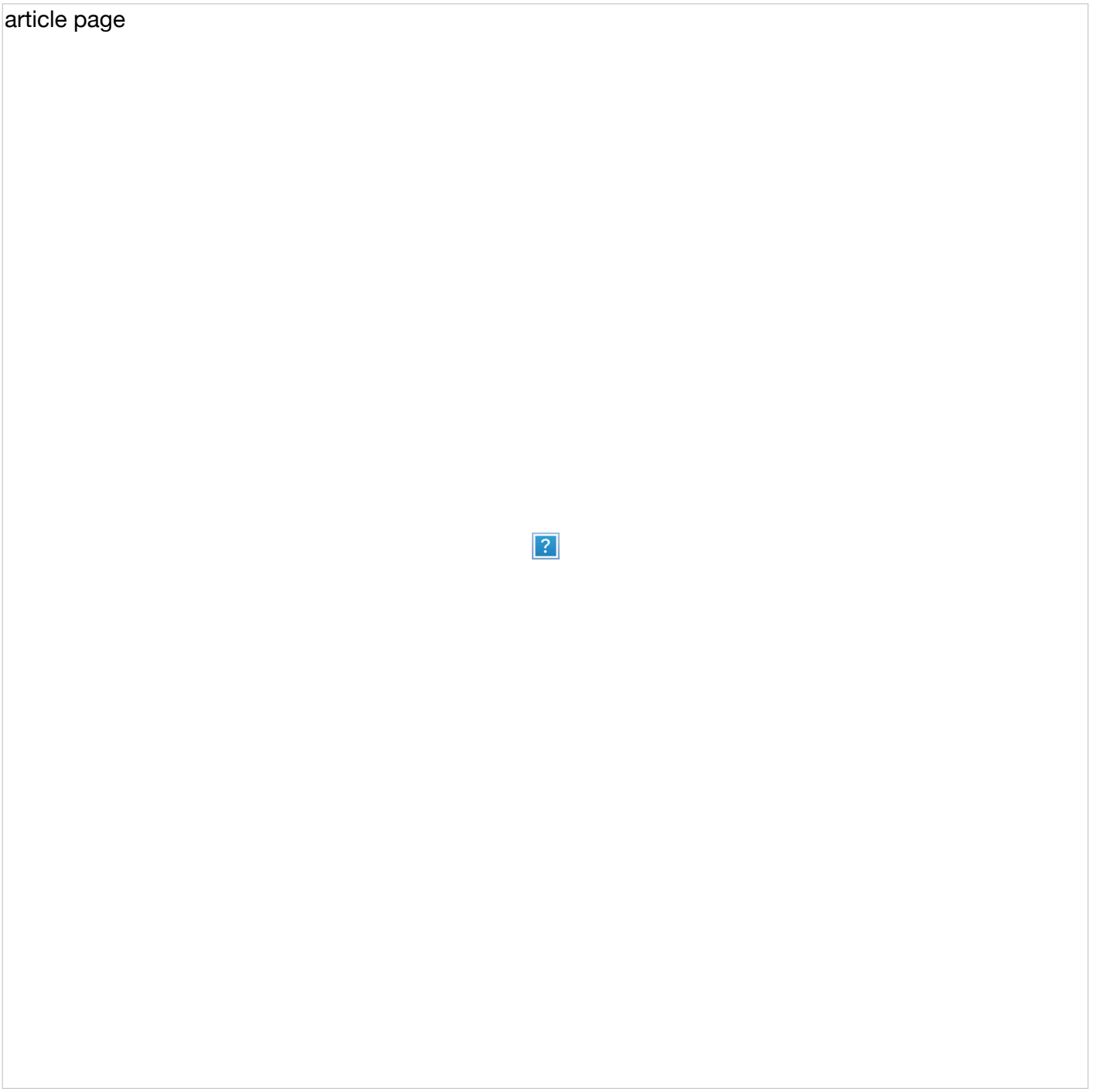

Figure: Distribution of word count in body texts

article page

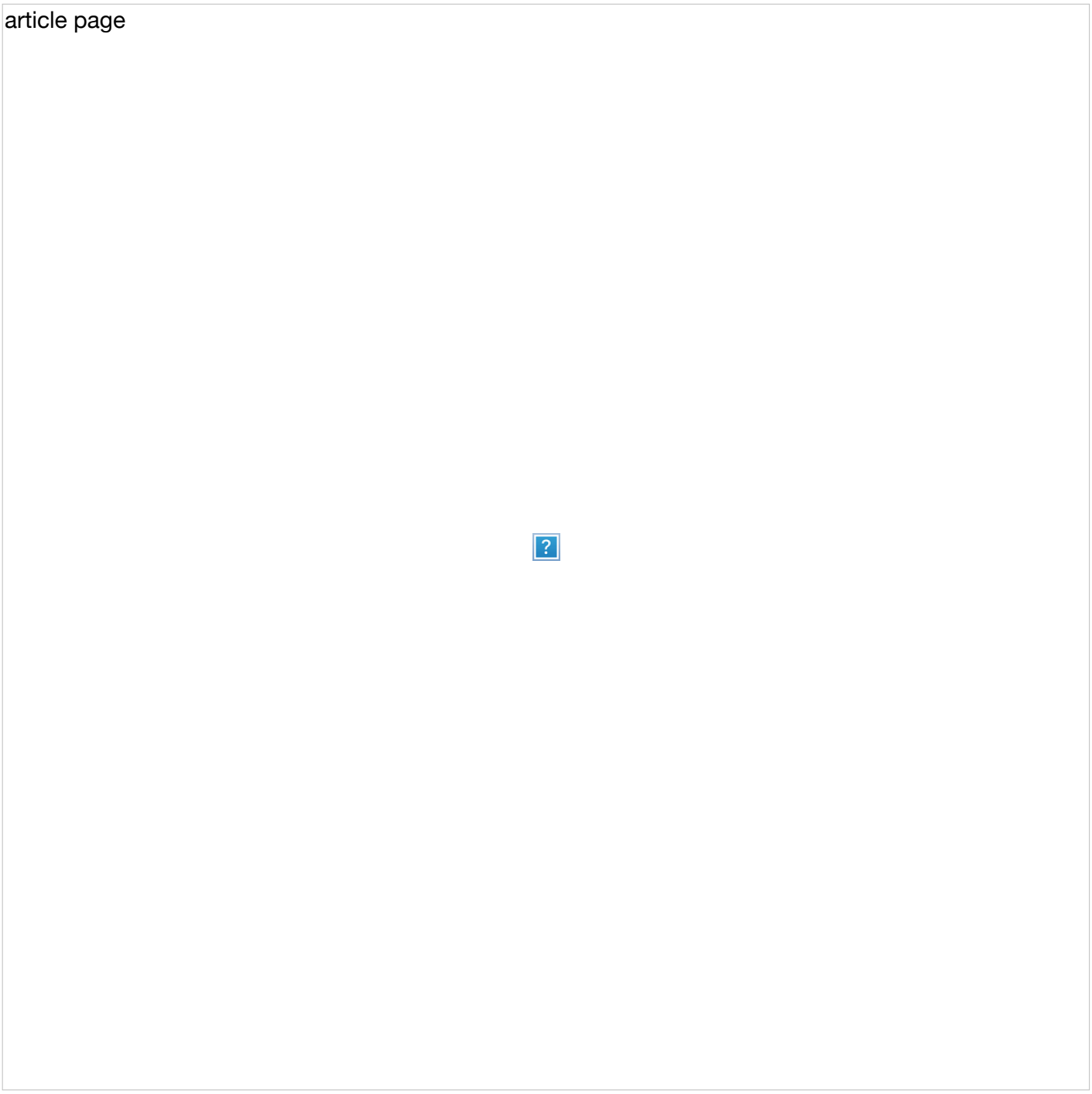

Figure: Distribution of weights in corpus

Conclusion:

- The disbribution of corpus words follows Zipf's law
- Considering the mean word lengths of documents, the number of documents (as per indication in the assessment outline) appears sufficient./li>
- Distribution of word count per documents appears within the expected range since the articles have different sources and represent different news types (daily summaries, brief summaries of events, extensive articles.
- For the NLP tasks, this means that
  - seen compliance with Zipf's law, the corpus is 'ok to go'
  - high variance in text types and lenghts is a challenge, which I approach by selecting only x keywords (not the full TF-IDF vectors) and y entities (once extracted)

Code

The code for this part of the assessment is available in the seperate file "A3_DocumentNumber_2_Code_sacha_schwab.ipynb".