Quartz Scraper Manual

This document is written to understand the usage of Quartz Scraper.

Quartz Scraper is a scraping engine that scrapes all the news articles uploaded to Quartz along with images, based on a search keyword.

For example, a search keyword of *flood* will download all articles with images from https://qz.com/search/flood/. The number of articles to download is configurable in the program.

The code can be found in https://github.com/ajaymaity/Quartz-Scraper.

Prerequisites:

Python 3

Pip

Chrome Webdriver (Make sure the driver executable is in PATH)

After the program is downloaded locally, perform the following steps to install necessary packages:

```
pip install requirements.txt
```

The above command installs all the necessary packages required to successfully scrap the website.

To run the program, execute the following command:

```
python scrap_quartz.py <search_keyword>
```

As an example, to scrap news articles related to *flood*, execute the following command:

```
python scrap_quartz.py flood
```

The above command starts the Chrome webdriver, scrolls the webpage enough to scrap 50 articles (the number of articles to scrap is configurable in the code), and downloads the images and news contents from each article.

The scraped news contents are stored in a CSV file: *scraped_news_content.csv*, whose structure is as follows:

```
ArticleURL, Tagline, Heading, Author, Datetime, Text, Figures
```

As a single article can contain multiple images, the `Figures` field above contains the range of figure numbers (such as 18-24) that references another CSV file, *scraped_fig_for_news_content.csv* that only stores figure details. Its structure is as follows:

```
FigureNo, FigureURL, FigureCaption
```

All the images are downloaded and stored in a folder *images* in the same directory where the program is stored and executed.

These images can then be filtered manually to build the train dataset based on categories such as Fallen Trees, Submerged Cars, or Submerged Signs.