

State Wise Stock Market Sentiment Analysis Using Web Scraping

WordCloud: A word cloud is a collection, or cluster, of words depicted in different sizes. The bigger and bolder the word appears, the more often it's mentioned within a given text and the more important it is.

Aim of the Project :

The project's main aim is state-wise stock market sentiment analysis for social media keywords planning and advertisement. This project uses web scraping tools to collect text-based data from platforms like Facebook, Twitter, Pinterest, and Instagram. The data is then processed using a natural language processing algorithm for specific keywords, as well as WordCloud for graphical representations of word frequency, which emphasize words that appear more frequently. The end output is a list of keywords that can be used for state-wise sentiment analysis of the stock market for social media marketing and advertisements.

Procedure:

a. Data Collection

1 Data Extraction using web scraping:

- The data can be extracted by running blogs.py and facebook.py on PyCharm or spyder IDE
- Requirements: chrome driver (link: <https://chromedriver.chromium.org>), install and import libraries like selenium, requests, by tesseract (OCR tool used for reading texts from images), pillow.

2 Running web scraping code:

- In facebook.py add your login id, password and keywords. In blogs.py add keywords and run
- Eg for keywords: learn stock market Bangalore, learn stock market Delhi etc.
- The web scraping code returns a text-based output as keywords.txt (eg: learn stock market Delhi.txt)
- Run the same codes for multiple keywords
- extracted text files are stored in, Karnataka text data and Gujarat text data

b. Data Cleaning

3 Merging the text files:

- use merging tools to combine all the documents
- merging tool: <https://products.aspose.app/words/merger/txt>
- merged text files are stored in merged text files

4 Importing the merged .txt file

5 Run the text file using keyBERT to get keywords and word cloud to get a visual representation of the frequency of the keyword.

6 Save the word cloud model to get keywords png image, word cloud images of both the states are saved in word cloud output images.

Web Scraping codes :

1. blogs.py

blogs.py is built using the pillow, tesseract, selenium web driver, and requests. It is an automated process that runs on chrome driver and is linked with the social searcher, which links to Google search, Facebook, Instagram, Twitter, and Pinterest domains. websites are opened in the order and data is extracted from all the mentioned platforms. the query parameter keeps updating based on the keywords related to search keywords, the maximum scrolling or website opening is set to 10, and tesseract is used to extract data from images. output is a text (.txt) file that consists of text-based data.

1. facebook.py

facebook.py covers all the libraries used in blogs.py. For web scraping, the user must provide a login id and keyword list. The keyword list may contain multiple keywords. We extract text and image-based data, including comments, user names, likes, shares, dates, and data from a user's post by using libraries. Pytesseract extracts text from images with a scroll limit set to 100 because Facebook has infinite scrolling. output is a text (.txt) file that consists of text-based data.

Working:

1. run facebook.py, the code logs in to the profile and collects data based on given search keywords, it collects data from posts and images
2. Facebook.py collect data from Facebook and blog.py collects data from various sources like blogs, Instagram, Pinterest Twitter etc. Both the web scraping tools run on chrome driver and it is controlled by automation text software selenium.

3. the output text file is combined using the merging tool
4. The merged text file is processed using the NLP algorithm KeyBERT.
5. KeyBERT provides keywords for sentiment analysis.
6. Wordcloud gives an image output consisting of multiple keywords based on word frequency.

Expected Results:

The result is a list of keywords that can be used for state-wise sentiment analysis of the stock market. By analyzing the keywords output by keyBERT, a user can better plan their social media advertising by understanding what types of keywords they need to use on Facebook, Twitter, Pinterest, and Instagram, as well as the way people from different states view the stock market, and by using WordCloud, the keywords are visualized based on word frequency extracted from the input text file. The word cloud (image) is used to understand the overall sentiment of an individual state towards the stock market in a single image whereas KeyBERT helps in determining the specific keywords that need to be considered before selecting keywords and designing advertisement creatives.

c. Model Development & Optimization

KeyBERT example:

WordCloud example:

Framework required to run this jupyter notebook:

- Python 3.6 and above
- Matplotlib for visualization
- keyBERT for keywords extraction
- WordCloud for graphical representation of keywords with high frequency
- Web scraping code blogs.py and facebook.py

Make sure that this is a template for reference only, copy pasting of the same will be not allowed from placement team.