Memelysis

Analytics of memes

Michał Dyczko
Jakub Kała
Maciej Pawłowski
Krzysztof Spaliński

Project description

The goal of this project is to prepare web-based dashboard consisting of:

1. Live stream of memes from various sources:
    a. Memes are automatically classified by category;
    b. Each meme contains information about it's score against other memes from that source.
2. Analytics of meme categories and formats popularity.

Data sources

Memedroid

reddit

imgur

Collected data

Raw scraped data:

1. Memes data in JSON format file;
2. Images in various formats;
3. Logs in text format.

Data preprocessing (before storing):

1. Using GCP Vision API to get text from images (OCR);
2. Filtering explicit content.

We predict to have ~200MB of data to be uploaded per hour.

JSON entry example:


```
{
    "url": "https://i.redd.it/xtvvohk5dnt41.jpg",
    "additional_data": {
        "date": 1587246469.0,
        "title": "Favourite Wii Sports Resort Game??",
        "upvotes": 70,
        "upvote_ratio": 0.97
    },
    "filename": "reddit_2020041900_00014.jpg"
}
```

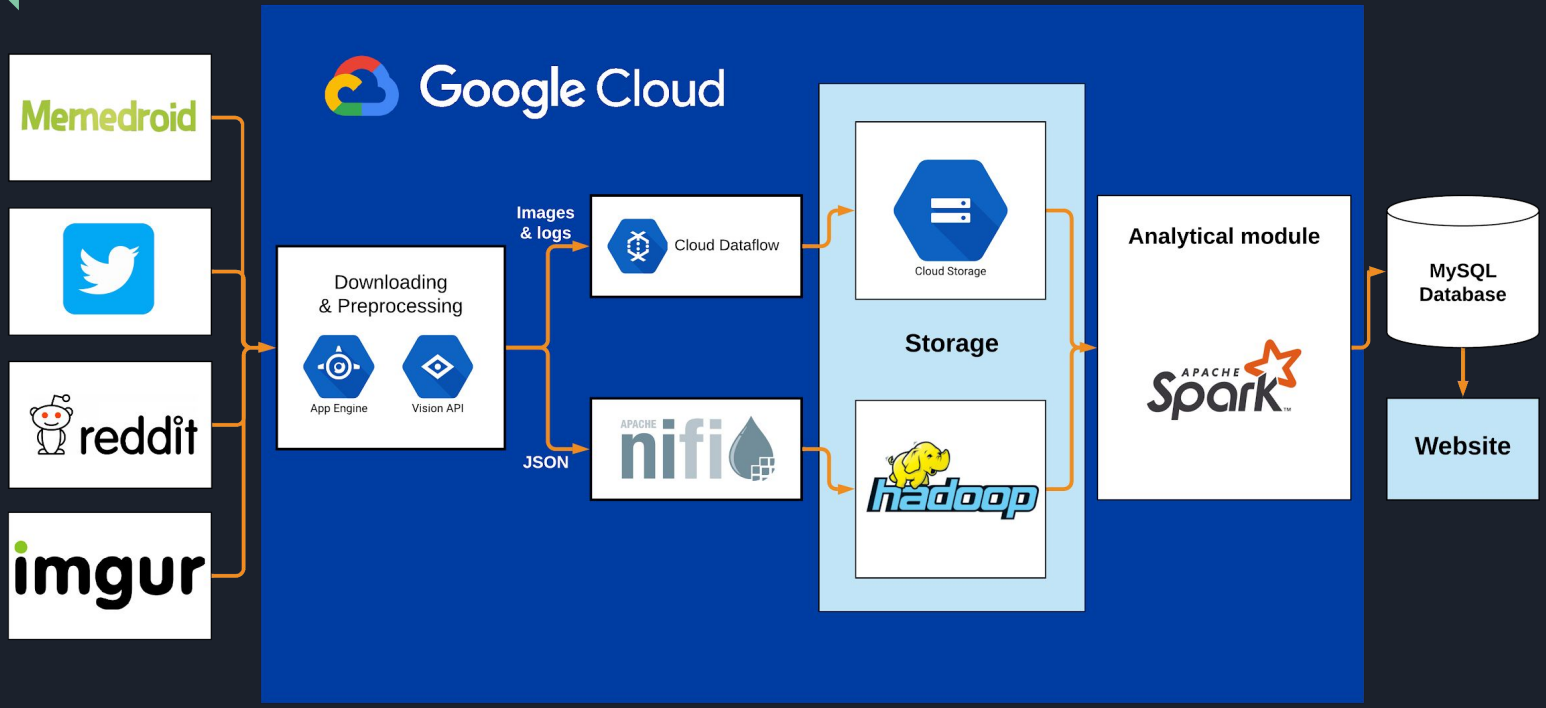

Log file example:

```
Downloading data from Memedroid on 2020 04 19, 11:40.
Scanned 120 memes, found 43 memes to download.
Downloading memedroid_2020041911_00001jpeg from
https://images3.memedroid.com/images/UPLOADED185/5e9a0e1ec6f40.jpeg...Done.
Downloading memedroid_2020041911_00002jpeg from
https://images3.memedroid.com/images/UPLOADED905/5e9a03bac0b81.jpeg...Done.
Downloading memedroid_2020041911_00003jpeg from
https://images7.memedroid.com/images/UPLOADED508/5e9a136fbc33f.jpeg...Done.
```

What’s inside preprocessed data?

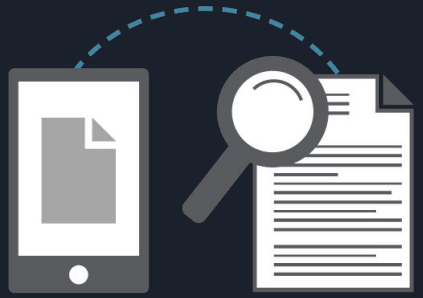
Source	Data	
	Source specific	Common
Twitter	• Tweet text • Hashtags • Retweet count • Favorite count	• URL • Filename • Text extracted from image
Memedroid	• Title • Date • Upvote ratio • Tags (optional)	
Reddit	• Title • Number of upvotes • Upvote percentage • Date	
Imgur	• Title • Tags • Up- and downvotes • Views	

Data architecture



Analytical module

Optical Character Recognition



Clustering

Virality factor analysis

Data processing

Batch processing:

1. Cluster analysis model re-estimation
2. Re-estimation of virality factor fitting model
3. Trends analysis

Stream processing:

1. Extracting text from images (OCR)
2. Filtering explicit content
3. Category assignment for each incoming meme
4. **Virality factor** computation

Tools

Scraping:

- Python libs: BeautifulSoup 4, requests, Django 3;
- API's: Twitter API, Reddit API.

Cloud - GCP:

- App Engine for scraping, image downloading and webapp hosting;
- Dataproc: Hadoop, Spark;
- Compute Engine - Nifi;
- Google Dataflow.

Analytics:

- OCR - Google Cloud Vision API.

Tests

Test log is attached in a report file.