Social sentiment analyser - draft 0.1

Large scale social networks content analysis framework for sentiment research

### MFF UK Software Project

*Project name: TBA*

*Team: Jan Pavlovský*

*Jaroslav Knotek*

[*Petra Marie Doubravová*](https://www.facebook.com/petra.m.doubravova?__tn__=%2CdC-R-R&eid=ARBjYfdZVEPpqr6XT23cs7VWrRUuvRnfJ5YoXyZ0gbtvt5L7MSJcxvd8N97skYVPpzsY3Z2vBnaBs31q&hc_ref=ARS-VzNvpn3s3mrBVKiuRE68YBajxGhf_yL4dOu6EV2aBmtjeU_ip_HYGQ99n73lCu0&fref=nf)*‎*

*Lukáš Kolek*

*...*

*Supervisor: Doc. RNDr. Irena Holubová, Ph.D.*

## Motivation and Goals

For more than a decade already, there has been an enormous growth of social networks and their audiences. People connect on these networks, post their life stories and opinions as well as pictures and videos, chat and form societies interested in common topics. Since the foundation of Facebook in 2004, there has been an countless networks founded. They are certainly and indispensable part of everyday life for majority of people nowadays, and affect us all.

As people post about their life and experiences, comment on other people’s posts and discuss all sorts of topics, they generate a tremendous amount of data that are stored on these networks. The amount of these data is enormous. And many have already figured out that all this data could be utilised in pursuit of commercial, public or personal goals.

As we already know from the example of Cambridge Analytica, networks users’ personal data can be used in, if not exactly harmful, very immoral way. The main issue with CA was that they were using personal data as marketing tools primarily in the 2016 US presidential elections. They were targeting people as individuals and assessing data specific only to them, that those people may not have agreed to share publicly.

If gathered in very large amounts and analysed statistically and, naturally, anonymously, there could be very useful patterns discoverable in all that data. And this is the path we would like to venture in this project.

Some data published online, especially on social networks, are publicly accessible and mostly general in character. However a lot of data have restricted access in some way. For example on facebook, most of people’s posts are visible only to their friends or friends of friends. And even if they are globally accessible, meaning anybody is allowed to see them, it is usually difficult to get to any relevant ones without any kind of connection.

There has been a number of projects already created in both, academic and commercial sectors. *(reference).*  However most of what can by openly found are quite narrow-purpose tools designed for a single specific task. What we would like to achieve is to generalise these tools and create a multi-purpose framework for gathering and analysing social networks’ content.

#### Use-case 1: sentiment analysis of elections (eg. for political marketing)

The task of sentiment analysis aims to determine the people’s attitude and feelings about a certain entity or topic, and model them in time. That way, we can find out which event affect the public sentiment the most and in which matter, and act accordingly.

Utilizing this approach, a politician could, for example, reflect on his actions and statements and see which have raised positive and which negative reaction among the crowd. From marketing point of view, he could learn useful lessons from that and assess his future direction accordingly.

#### Use-case 2: market sentiment analysis

When referring to stock or financial market, we can analyse and model sentiment about a specific company stocks, or a currency. That way we can research behaviour of traders in reaction to certain stimuli. We can compare patterns in historical sentiment model with the graphs of actual market of stocks, as well as with events and news in the outside of the market. If we would be able to find correlation between these, we could possibly predict the development of market in future.

This can be especially useful with markets with high levels of volatility and manipulability (and therefore practically very manipulated). A first-hand example of such fast-paced and highly manipulated market are the cryptocurrencies, and their trading platforms.

#### Real-Time prediction

Processing historical and present data are different fields though. Modelling historical sentiment development can be done with batch-processing, using data gathered over long period of time.

To give an example, if we want to use sentiment analysis for trading decisions, we need to react quickly to current event. For that, we need the system to be able to process the data in real-time, as streams. Besides primary technical challenges of developing such a system, there are several additional issues, among those: limited network bandwidth, social networks limiting access for single access token and computational power. Nevertheless, the benefits of being able to process the social networks content in real-time and predict events development based on that could show to be quite far-reaching.

Share

File

Edit

View

Insert

Format

Tools

Add-ons

Help

Last edit was made 3 days ago by Jaroslav Knotek

Normal text

Calibri

Editing

## Overview

A system of this kind would generally consist of 4 primary modules:

1. Data acquiring
2. Pre-processing
3. Storage
4. Analysis

### Data Mining

Data mining is the process of gathering the data for future use. There are a few distinct tasks necessary to carry out in this phase. From high level view, these would be:

* Social network access (access tokens, accounts, access limits…)
* Data crawling
* Data downloading

### Pre-processing

Appropriate data needs to be firstly selected and sorted, before storing in the database, so that we do not store unnecessary data. Also, we want to store only relevant data to their specific topics and, in ideal scenario, store only the minimal necessary amount of data.

The other important task to do here is the data transformation. Especially when we want to aggregate data from multiple sources, and possibly also different forms of data (text, images, sound…), we need to transform the data into common form, so that it can be stored, accessed and read in a generalised way later.

### Storage

There are many options for storing structured big data available on the market and utilised today. The decision about specific technologies and approaches to use here are mainly affected by the specific use-cases they are aimed for.

Important factor to consider here is correct categorisation and deduplication of data. As a single post or image could easily relate to hundreds of topics and entities, we need to store the data in a way such that they do not get duplicated in the database, but also are kept linked to all entities they should.

### Data analysis

When we have all the relevant data stored in a structured manner, we can finally advance to actually analyse it. There have been many methods proposed and used for statistical analysis of big data, both conventional and machine learning.

The choice of specific technology and method here is again very use-case dependant. That is why we would ike to tackle this module in the most general way. The plan is to design an efficient and easy to use interface from the data storage, so that various data analysis tools may be used, and possibly even aggregated to solve a single task as effectively as possible.

## Time and tasks estimation

1. Analysis (cca. 1 month)

In the beginning, we need to state more specifically what should be the output of our work, assess our ability to carry it out, and carefully specify what if achievable in the time frame given. We also need to study existing materials about the topic in this phase.

1. Design, work division (1 – 2 months) – Following the specification, we as a team should make fundamental design and architectural decisions about the software. Based on that, we can divide work among team members.
2. Implementation (4-5 months)

Individual team members will implement assigned modules of the system. We would like to adapt an iterative development approach, in which we would iteratively integrate and test individual system modules.

1. Finalisation, documentation (1-2 months)

In the terminal phase, we will put together the final product, assess what has been accomplished and what hasn’t. Also, we will create the necessary user and development documentation.

--- JK version bellow

# Problem

Social networks provides abundance of information concerning various topics in form of posts, comments, and other. Amount of posts and comments is raising so quickly that no human is able to read them all and get overall opinion. Social networks do not offer tools supporting quick orientation in the overflow of posts.

The goal of this project is to create framework computing sentiment analysis over data published on supported social network for a given period of time. Result of the analysis will be present to user with structured format consisting of

* sentiment analysis
  + in time if possible
* significant keywords
* clustered opinions
  + positive, neutral, negative

# Social networks generally

Supported social sites are

* Twitter
* Reddit
* Facebook

Social networks contains abundance of data, but does not give them up easily. Social networks does not allow for querying extensive data. Data acquisition is limited(for a free user) by requests per time period (15 minutes for twitter, 1 hour for facebook) and amount of record retrieved. Those limits makes it impossible to support extensive online analysis.

Lack of data can be mitigated by following approaches:

* Analyse only data spanning over short period
* Use offline analysis - watchdog

# Data retrieval

Short period analysis can be made effective using various query filters but this will work for thousands of post but not millions. On the other hand, employing watchdog can process millions of posts but in very long period of time.

User can decide that it is important to analyse data from a given date onwards. This application will then start fetching newer post until all posts are retrieved and then it will periodically watch for newly added.

In both modes, data come in batches. In case of short analysis, data are processed after all of them are loaded and analysed are after. Watchdog is designed to perform analysis after each batch and update overall result.

# Data storing

*Short time analysis* and *watchdog mode* require data to be stored in a given structure. Each social network has unique structure of data which has to be unified into single form and then stored into common storage.

Unified form has to reflect metadata about post and anonymized data about user. Important for analysis are:

* Text
* User
* Keywords

To prevent keeping of obsolete and uninteresting data, anything older than a given time gets deleted.

# Sentiment analysis

# Use cases

In general, user selects data source which all data are acquired from, and then it select what type of analysis will be performed.

In case of social network, user selects for example twitter as a source of analysis. Then he supplies credentials and topic in which he is interested int and selects sentiment analysis. This job definition is then translated to a command chain beginning from data acquisition to analysis and then to visualization. If it is a long-running job, user is given a special url which will store up-to-date result. When job finishes result can be saved.

# Architecture and technology

Software consists of four cooperating modules

* Data acquisition - custom made module acquiring data from selected data sources.
* Analyser - custom made module analysing stored data according to selected analysis type.
* Storage - custom made wrapper encapsuling existing database solution
* Coordinator - custom made module coordinating work of all previous modules from job definition to data acquisition and transformation to analysis to visualisation.

It will be implemented as a web application. Various architectural patterns may be used. Simplest plausible model is to implement four Web services. It is easy to create and easy to manage. Main disadvantage is, that it does not scale very well.

Better performance will be achieved by creating finer grained modules encapsulated in containers and then orchestrated according to defined rules and performance requirements. This approach is much more complex, but if managed well, almost no management is required, and scale according to data load. Scaling is expected to be solution for long-time running analysis module. It won’t be required by *Data acquisition* module since it can’t load too much data.

Further analysis is required for proper technology selection.

# Extensibility

Application is designed to obtain data from social network in small batches and then store them. Stored data can be then loaded and analysed. Although, additional functionality can be added using user-made modules.

*Data acquisition* module can be extended by adapters to various data sources: file, other social network. Only requirement is that data must follow given schema.

*Analyser* module can be extended by custom made analysis. This alternation must come with appropriate visualisation tools. This project will not support

# Experiments

We will also experiment with multiple accounts which can speed up data acquisition.

Scaling