**POLITECNICO DI MILANO**

Department of Electronics, Informatics and Bioengineering

M.Sc. programme in Computer Science and Engineering



**Automated Analysis of Social Data**

**using Machine Learning Techniques**

Advisors:

Prof. Letizia TANCA

Prof. Maristella MATERA

Ing. Riccardo MEDANA

Master Thesis of:

Petar KORDA Matr. 852627

Pavle VIDANOVIC Matr. 854472

Academic Year 2017-2018

*We would like to express our sincere appreciation to our supervisors and our*

*families for all their support. We would also like to extend a special thank you to*

*Prof.ssa Maristella Matera for her time, guidance and understanding.*

**Abstract**

In today’s society everything is happening on the internet, in particular on social networks. Social networks play central role in everyday life of average person. So naturally, companies recognize the opportunity and try to make use of that by changing their business plans and focus to potential customers on social networks. Business is realized by company’s presence on web and producing a content that will take customer’s attention. In return users share their opinion about particular products by leaving comments on them and reacting on company’s posts.

Taking that into consideration, it is useful to have an automated way to check user reactions on products that company is offering. Also knowing types of people fallowing and leaving opinion on products can be turned into advantage for creating future business plans. For example, to predict which products can be attractive for specific user groups or to determine best time when to lunch products. We recognized the potential of that and that’s why we were eager to examine sentiment analysis tools and machine learning algorithms to achieve that goal.

In this thesis we have built automated for calculating sentiment of users who commented on specific company’s post along with intelligent spam filter. Sentiment analysis was done separately on text and on the emojis. For the evaluating text sentiment, we used open source API and for emojis we used table of evaluation for each emoji. Spam filter was designed using supervised machine learning techniques to determine spam, not just by searching URL patterns in comments, but also to determine the spam by checking text content. We have also built clustering module which uses unsupervised learning techniques on user data and visualizing characteristics for each discovered group. Finally, in last chapter is defined how previous models can be used together in an API to evaluate success of company’s posts.

**Contents**

[1 Introduction 8](#_Toc497680533)

[1.1 Structure 9](#_Toc497680534)

[2 State of Art 10](#_Toc497680535)

[2.1 Need for analysis of social data 10](#_Toc497680536)

[2.2 Applications of social data analytics in various companies 10](#_Toc497680537)

[2.3 Methods and tools used for social data analytics 11](#_Toc497680538)

[3 Automated sentiment analysis of social network content 12](#_Toc497680539)

[5 Unsupervised approach to Social Data Analysis 14](#_Toc497680540)

[5.1 Fetching data from social networks 14](#_Toc497680541)

[5.2 Pre-processing 14](#_Toc497680542)

[5.3 Unsupervised learning techniques 14](#_Toc497680543)

[5.4 Elbow analysis 14](#_Toc497680544)

[5.5 Visualization of clusters 14](#_Toc497680545)

[6 API 15](#_Toc497680546)

[7 Conclusion 16](#_Toc497680547)

[Bibliography 17](#_Toc497680548)

[List of figures 18](#_Toc497680549)

[List of tables 19](#_Toc497680550)

**Chapter 1**

# **Introduction**

Analysis of social network content is difficult because conversation on social networks differs in many ways from normal conversation. Contents are rich with emojis, hashtags, mentions and spams which the one need to filter and process along with raw text to find the information behind it.

By analysing social networks data about certain products, brands or certain campaign can be very useful for companies and their business. Companies can predict future trends, increase the profit and thus be in advantage over the competition.

Within previously described context, this project gives opportunity to user to analyse contents using sentiment analysis to determine sentiment of users on specific product, supervised machine learning algorithms within spam filter module to efficiently detect and remove spams from dataset and finally clustering module that discover user groups and their characteristics which can be used in making future predictions.

Sentiment analysis mentioned before represents the process of computationally identifying and categorizing opinions expressed in a piece of text, especially to determine whether the writer's attitude towards a topic, product, etc. is positive, negative, or neutral [Oxford dictionary definition]. It is combined with emoji sentiment evaluation in a way that sentiment analysis is done on the raw text using open source API while the emoji’s sentiment is evaluated using sentiment tables. Final sentiment is defined as specific combination of those two sentiments.

Spam filter module is equipped with two components. First part is done as text processor using regex expressions to detect links inside the text. Second part is trained Naïve Bayes machine learning model for detecting spams by checking the text context and represents more intelligent way of doing it.

Clustering module is created as unsupervised machine learning model, that taking users finds optimal number of user groups. It is also equipped with visualizing part that displays specific characteristics for each user group.

Finally, previously described models used together can be used in one complete API which will make predictions on successfulness of company’s posts left on social networks. That API can be used as a part in any company’s business intelligence application.

## **1.1 Structure**

* Chapter 2 is about the reasons for having tools for analysing social networks within company’s applications, the advantage of having it and how to use it. We will mention examples where we can use it.
* Chapter 3 is dedicated for automated sentiment analysis of social network content. Here we are going to talk about sentiment calculation, pre-processing of emoji’s and their sentiment, and combining text emoji sentiment. We are going to describe our approach and architecture.
* Chapter 4 is about machine learning approach to create spam filter. This includes pre-processing, training the dataset and evaluation of classifier.
* In chapter 5 we will talk about unsupervised approach to social data analysis. We will go through all steps of this process which includes: fetching the data from social networks, pre-processing, unsupervised learning techniques, elbow analysis and visualization of clusters characteristics.
* Chapter 6 is reserved for describing API which will use previously mentioned modules in order to analyse custom social data.
* Chapter 7 will finalize the purpose of this project and present the conclusion. We will also mention future improvements of this project.

**Chapter 2**

# **State of Art**

This chapter describes state of art of analysis of social data using machine learning techniques. Chapter consists of three sections, each of them trying to emphasize the need for analysis of social data and their impact to company’s business.

1. Need for analysis of social networks data
2. Applications of social data analytics in various companies
3. Methods and tools used for social data analytics

## **2.1 Need for analysis of social data**

Social media gives businesses an unprecedented opportunity for connecting with customers and prospects. While there are numerous social networks that provide you with a vast array of tools for providing customer service, explaining how your products work and much more, it’s important to realize that simply having a social media presence is no guarantee of success. It is essential to test and track your results so that you can discover the most effective strategies and that is why analytics of social data are so important.

We know that users of social networks represent protentional customers, they are also proactive in a sense that they can share their opinion about certain products that company or brand shares. So, in order to grow their business and to increase the profit companies need to analyse user reactions and behaviour on their products shared on social networks. As a result, company can have model of a user and can predict weather new product will be successfully accepted and attractive to potential customers.

## **2.2 Applications of social data analytics in various companies**

Social data analytics can be applied in many types of companies. Especially the ones which includes marketing and where the competition is big. So, when there is strong competition it is very important to be innovative, but also successful in the same way. This is done by analysis and predicting the successfulness of new products. For example, social data analytics can be used by fashion industry brands to see the reactions of their followers about new products. They can also use to see which types of user groups their followers belong to make wright predictions in the future. Fashion companies can use it to decide when is the best way to lunch specific products. It also depends are the followers young people like students, mid-age or older people, which of two genders are reacting better and what to do to stimulate their better reaction.

## **2.3 Methods and tools used for social data analytics**

Some of the most famous tools that have been used for social data analytics are Sprout social, Buzzsumo and Google analytics. Sprout social can measure performance across Facebook, Twitter, Instagram and LinkedIn, all within a single platform. Having analytics at one place makes it easier to track and compare your efforts across multiple profiles and platforms. It is recommended for any brand that manages multiple social media profiles across multiple networks. Buzzsumo is different than the other social media analytics tools on our list. Instead of analysing your brand’s individual social media performance, Buzzsumo looks at how content from your website performs on social media. For instance, if you want to see how many shares your latest blog post received on Facebook and Twitter, Buzzsumo can provide you with that data. Buzzsumo will not only show you the number of shares for each piece of content, but it also shows you which type of content performs best on each network based on length, type, publish date and more. Google analytics is not technically a “social media analytics tool,” Google Analytics is one of the best ways to track social media campaigns and even help you measure social return on investment. You likely already have GA setup on your website to monitor and analyse your traffic.

All mentioned tools use statistical techniques for analysing the social network data. While they are analysing user reactions, they don’t support analysing contents left by users, such as comments and posts.

For that part it is useful to have some machine learning techniques for analysing text context and that is what we recognize and try to create complete API that will include all modules together.

**Chapter 3**

# **Automated sentiment analysis of social network content**

**Chapter 4**

**Machine Learning Spam Filter**

**Chapter 5**

# **Unsupervised approach to Social Data Analysis**

Social data represents data about the users like his gender, city, description, date of birth etc. Those data can be used to detect similarities between the users. Some users are more similar and can be identified in the same group. So, the main goal of this module is to find those groups of users and to visualize their characteristics.

## **5.1 Fetching data from social networks**

To be able to do analysis of user data, first we need to have that data. User data are present on social networks but to gather it, we need to have right permissions. For the purpose of this project we were provided a dataset which included comments, posts, social channels, brands and desired social networks with fallowing relationships:

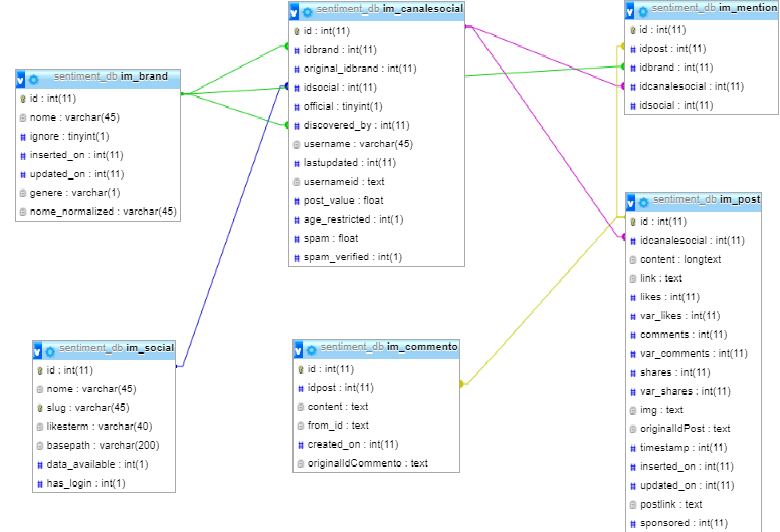


Fig1 Sentiment\_db

So, from the graph we see that for every post from *im\_post* table, we can fetch all comments that are located in table *im\_commento*. In Chapter 2 is explained how to determine summarised sentiment of all users about the post. Now, we are going further buy determining what are user groups that comments, we are interested in their characteristics and behaviour of each group. To get user data we need to use API for each social network that we are interested in. We will explain how Facebook API is used for this purpose and for the rest it is very similar procedure.

Facebook Graph API requests user authorization access before applying search. User connects to Facebook API by sending access token and to get it user need to register at *developers.facebook.com* and to register the application that will use the search. After successful authorization then user can do the search.

Id parameter is target user id and the other is list of user attributes that we are interested in. The problems we met are related to permissions and user privacy options, since not all users shared all desired attributes. So, our dataset is not full, and we used some pre-processing techniques to overcome it and that is explain in next part of this chapter. The result is return JSON formatted string. All the data fetched are stored into *user\_social* table.

{

"birthday": "01/01/2010",

"about": "Fashion, Trends, Beauty Style, Personal Shopper and Stylist.",

"email": "info@driferreira.com",

"city": "San Diego",

"country": "United States",

"category": "Personal Assistant",

'description": "Dri Ferreira is a Stylist and Personal Shopper for the Fashionable Elite for over 10 years. Get Daily Inspiration from Me & My Team"

}

Listing x.x: FB API response

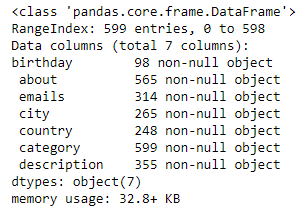
## **5.2 Pre-processing**

Getting the target data is one thing, but getting the information behind that data is something else. Raw data is usually need some modifications, transformations or in one-word data needs pre-processing.

In every project related to data science and machine learning pre-processing step takes almost 60% of time.

### **5.2.1 Exploratory data analysis**

First thing that is usually done in pre-processing is exploratory data analysis. We are inspecting features (columns) of the dataset. We are inspecting basic statistic, types, visualize distributions, correlations etc. In the next picture there is example from user data who commented on specific post:



Listing x.x: *user\_social* info

 Also, data types can be numeric, categorical or string variables. Here we must pay great attention to string variables, since they are unstructured rich with hashtags, links and emojis. Also, content is generated on many different languages, so we need to translate it in order to have useful information. We can see example on next dataset.

Figure x.x: *user\_social* table

### **5.2.2 Missing data**

Datasets that are collected directly from users usually are not 100% full. We have to handle missing data in one of many ways following certain policy. For example, we chose to discard rows that have more than two missing fields. For the rest of the rows we fill missing fields using policy depending on the type of the field. For features that are:

* Categorical – we fill the missing value with most frequent category in specified feature
* Numerical – we selected median value since it is more resistant to outliers than median

In our dataset, all the variables was categorical, and we choose to fill-in missing data following distribution of values.

### **5.2.3 String data**

String variables need to be handled in special way. We decided to transform them in categorical variables by fallowing next steps.

First, we need to have language consistency, so we used *Google Translator* to translate text to English. After that, we tokenize string by removing punctuations and stop words. Now we have list of tokens instead of the string. Next, we create dictionary, where we count occurrence of each token and select top 10 since they take most of the variance of the dataset. Finally, we go again threw all rows and calculate similarity between the tokens and categories, and select most similar category. In that way we did categorization of this string feature. We did this categorization on about and description features.

The birthday feature which was also stored as a string was split into three new integer features: day, month and year.

### **5.2.4 Categorical data**

Categorical data are variables that often contain label values rather than numerical values. The number of possible values are limited to fixed set.

Some examples include:

* A “pet” variable with the values: “dog” and “cat“.
* A “color” variable with the values: “red“, “green” and “blue“.
* A “place” variable with the values: “first”, “second” and “third“.

Categorical data are perfect choice for some type of algorithms like decision trees. Although, there are also some algorithms that cannot work on label data directly, they require input and output variables to be numeric. This means that categorical data needs to be converted to a numeric form. Method for doing transformation is also used in this project has two steps:

1. *Integer encoding step* where each category is assigned an integer value. For example, “red” is one, “green” is two and “blue” is three.
2. *One-hot encoding* is necessary for categorical variables where no such relationship exists. One-hot encoding is applied to integer representation of categorical variable. This is where integer encoded variable is removed, and binary variable is added for each unique integer value. One binary variable is redundant an thus it can be removed. Final binary variables are called *dummy variables*.

## **5.3 Unsupervised learning techniques**

Unsupervised machine learning is machine learning task of inferring a function to describe hidden structure from “unlabelled” data. Classification or categorization problems are not included in this observation. Since the data in those problems are not label it is not possible to measure accuracy of the model outputted by the relevant algorithm, which is one difference between supervised and unsupervised machine learning algorithms.

**5.3.1 Hierarchical clustering**

Hierarchical clustering is one possible type of clustering. It can be agglomerative or divisive. Agglomerative clustering starts with all individual points as separate clusters and then starts merging them by similarity which means in each step we merge most similar clusters until we get one cluster. Divisive clustering goes in other way it starts with one cluster and at each step it separates into more similar clusters until it reaches number of clusters as there are data points. With this method we use dendrogram to visualise clustering and choose best number of clusters. Advantage of this method is that we don’t need to know number of clusters in advance and the disadvantage is complexity of the algorithm which is time exhausting for big datasets.

**5.3.2 K-Means**

K-means clustering is other type that we applied in our project. Before we use it, we need to specify number of clusters that we expect to have. In start, random centroids which number we specified are initialized and other data points are assigned a cluster which is the closest centroid. Algorithm is specified in fallowing figure:

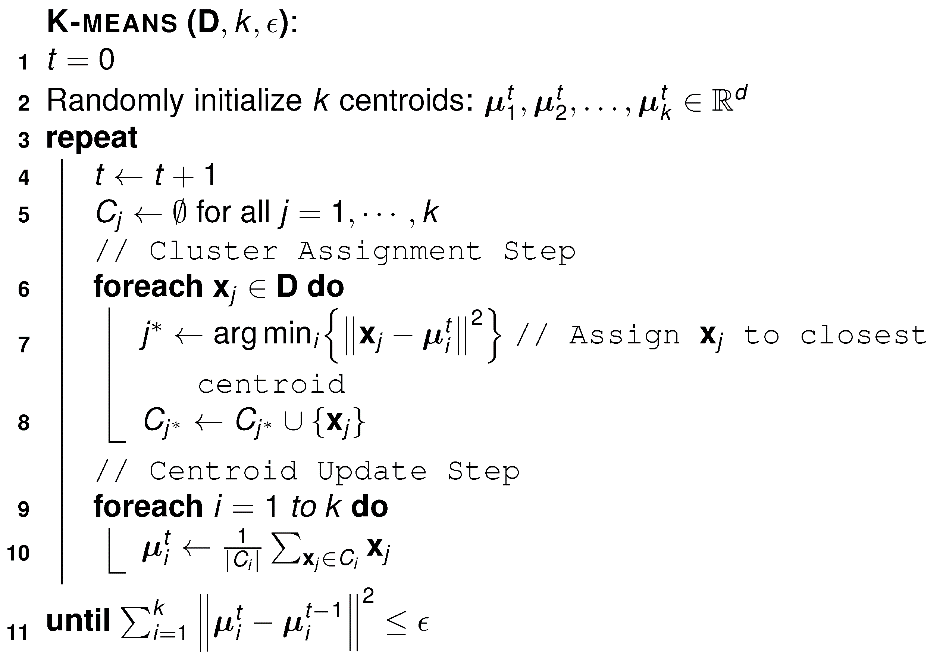


Fig x.x: K-Means agorithm

Advantage of this algorithm is that it is computationally fast algorithm and also it produces tighter clusters, especially if the clusters are globular. Disadvantage is that we need to know number of clusters in advance and it is not good for clusters with non-globular shape or with variating density . This algorithm had the best performance on our data, so we chose it for our dataset. Number of clusters was determined using Elbow analysis method which will be explained in fallowing subchapter.

## **5.4 Elbow analysis**

As we mention before we have chosen K-means clustering technique to inspect our dataset. It had the best performance, but we need to find best number of clusters. So, to be able to do that we need to measure performance for each trial number. We used two measures:

* Within-cluster sum of squares (WSS):

where is the centroid of cluster (in case of Euclidean spaces)

* Between-cluster sum of squares (BSS):

where µ is the centroid of the whole dataset

For each number of clusters, we are making WSS/BSS trade-off. We can plot this to and use elbow method to determine optimal number of clusters. We can see that on next figure:

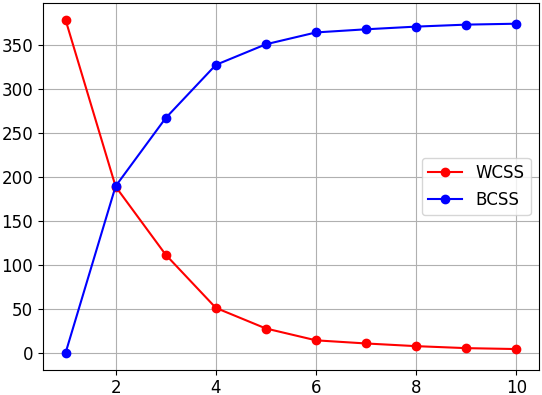


Figure x.x: Knee for K-means clustering

From the graph we need to find one point after which WSS is stabilized. In our case, the x-axis represents number of clusters and we can see that after point representing four clusters, WSS is not changing much i.e. it is stabilized. So, in our case optimal number of clusters is four.

## **5.5 Visualization of clusters**

**Chapter 6**

# **API**

**Chapter 7**

# **Conclusion**

# **Bibliography**

# **List of figures**

# **List of tables**