Using Facebook to predict election results

Andre Faria[1], Prof. advisor - Rui Fuentecilla Neves

Instituto Superior Tcnico, University of Lisbon, Portugal
andre.lince.faria@tecnico.ulisboa.pt,
tecnico.ulisboa.pt/en/

Abstract. Social Network sites like Facebook have been widely adopted by politicians in election campaigns with the objective of increasing the potential electorate. The purpose of this paper is to propose and implement a new algorithm for automatic prediction of political orientation of Facebook users based on their social activities by comparing them with activities from Political Parties on the same platform. The usage of *Posts*, *Likes* and *Comments* from Political Pages and from the user social activity, a machine learning classification algorithm will be built with the capability of associating a user as a supporter of a specific political party.

Keywords: Social Networks, Facebook, Machine-Learning, Classification, Automatic political profiling, Data mining, Text mining.

Index

Using Facebook to predict election results 1
Andre Faria, Prof. advisor - Rui Fuentecilla Neves
1 Introduction 3
1.1 Overview 3
1.2 Objectives 3
1.3 Outline 3
2 Related Work 4
2.1 Social Networks 4
2.2 Algorithms 6
2.3 State of the Art 15
2.4 Conclusions 20
3 Solution Proposal 20
3.1 Architecture Overview 20
3.2 Phase 1 - Data Pre-processing 21
3.3 Phase 2 - Data Classification 22
3.4 Integration and Execution 25
3.5 Expected Results 25
4 Evaluation Methodology 25
5 Calendar 26
6 Conclusions 26

1 Introduction

1.1 Overview

Social networking is a way for one person to meet up with other people online. Some people use social networking sites (like Facebook or Twitter) for meeting new friends, reconnect with old ones and to share interests and opinions. As such, these platforms have become and enabler for human interaction and are increasingly becoming a fundamental cornerstone on which our society depends on. Recently, social platforms have been exploited as a source of social capital[1][2][3], especially for career promotion and marketing campaigns by private users and enterprises[4][5]. In addition users employ social networks for political discussions and communication[6].
Politicians began to realize the potential of social networks for empowering their campaigns and their public image[7][8][9]. Many political party strategists started to use Facebook as a backbone for their party campaigns on which they attempt to influence professional journalists, reporters and appeal to supporters[10]. Recent studies tested the possibility of predicting the orientation of a Facebook user based on their likes and posts in comparison to political party standards and have had a surprising success when verified against real results[11][12][36]. Making an accurate prediction, in a political perspective, is of high importance to the political parties as if these forecasts can be believed as precise, working on a re-planning and readjustment strategy, may influence the people on the voting process to vote in favor.

1.2 Objectives

This work expects to aggregate and analyze all information available on Facebook believed to be effective by proposing an extended way of predicting the political orientation of a user by considering the user *Likes*, *Posts* and *Comments* of such user on political party pages as well as in his Profile and Feed pages.
Being able to discriminate a user as a specific political party follower by his *Posts*, *Comments* and *Likes* allows the algorithm to extract relevant features of each group of users that fall into that party as supporters. By extracting such features it's possible to create an inference based prediction for users where no such information like *Posts* and *Likes* are available, extending the range of the algorithm to users who dont have that much data publicly available. The developed algorithm should be able to make a more accurate prediction on user political orientation than previous attempts and above all to grasp with high accuracy the Portuguese election results coming in the first of October of 2017.

1.3 Outline

The outline of this paper is the following:

- **Chapter 1** makes a short introduction about the work at hand.

– **Chapter 2** introduces the reader to the topic of Social Networks, outlines a few relevant machine learning algorithms, and finally explores some works which directly correlate to this one.
– **Chapter 3** exhibits the proposed solution by providing a step-by-step explanation about the algorithm.
– **Chapter 4** proposes the evaluation methodology to be followed throughout the development phase of the solution.
– **Chapter 5** contains the calendar relative to the development stages for the solution.
– **Chapter 6**, the final chapter, withdraws some conclusions about this paper.

2 Related Work

In this section I will mention some works and methodologies that inspired the development of this thesis. First i will explore the impact of social networks in today's world and how that has influenced and moved people. Next i will introduce some machine learning algorithms that i'm considering for use when developing the final solution. In the course of development i predict that some adjustments regarding those techniques may change but still every deviation from what was planned is important to be expected. Finally, in the closing topic of this chapter i will analyze a few works done by fellow scholars that encouraged me to pursue this path of politic related affairs.

2.1 Social Networks

Commonly referred to as virtual communities or profile sites, a social network is a website that brings people together to talk, share ideas and interests and make new friends. Today, social networks are used not only for social purposes but also for business purposes or both, being that the trend is for them to keep growing thus becoming essential. Nowadays these platforms offer a wide variety of services as they are so many and so diversified. The most remarkable and distinguishable are Facebook, MySpace, Twitter and LinkedIn each of them having an estimate from 400 million to 1.2 billion users whereas Facebook tops roughly 2 billion[13]. Social networks pioneers include hi5 and the first social network of the masses: Friendster[14].

Facebook

The largest visited social network in the world[15] was founded in 2004 by Mark Zuckerberg, Eduardo Saverin, Andrew McCollum, Dustin Moskovitz and Chris Hughes [16][17]. When Facebook first appeared (named thefacebook) its initial purpose was to serve as Harvards internal social network with restricted access from the outside. Breaking down Facebook to a set of components results in the following topics: Profile, News Feed, Friends, Pages, Posts, Likes.
Note: this breakdown could be applied to any other social network in similar fashion.

Profile

The users profile page is where all the information that defines that specific user is at. The most noticing features are his user name and profile photo that allow several people (also registered) to get a glimpse of who and what that user is about. All data that directly relates to the user, his education, who/what he/she likes and befriends is depicted in *Profile* page.

News Feed

The news feed is the page where all new updates are displayed. The user only sees what he or she chooses to see. For example if Im friends with some other person or if I follow my football club activity, information updates about them and them only, appear on the News Feed. One point to keep in mind is that in this feed a user can view the public activity of his friends and if his friends activity intercepts his own he will be notified about it enhancing the user experience.

Friends

A social network only exists if there are other people besides yourself, people you like and have interest on, and that make you go there in order to know whats new about them. If a user had no friends or contacts to whom he is connected with, the social network purpose would not be achieved. Note that the *News Feed* described before also includes recent activity made by the users friends.

Pages

A user can create Facebook *Pages*. *Pages* are used to advertise something. They can advertise a project the user is developing, can be used for sharing ideas and can also be used like a profile, with a name and picture, but with the exception that the page itself cant have friends; only followers, people that demonstrate interest on what the page contains.

Posts

Posts are nothing more than an activity the user executes which content is displayed to others. *Posts* related to a user can appear in one of four places: his *Profile*, other users *Profile* page, the *News Feed* or in a specific *Page* (as described before). *Posts* represent an opinion meant to be shared. If as a user decide to make a *Post*, besides expressing myself Im allowing other users to express their opinions in regard to my *Posts* content. Other people can express themselves by using the option *"Comment* which is one powerful feature as it directly relates other peoples opinions to mine.

Likes

Likes are perhaps the most direct and meaningful feature Facebook and any other platform offers. In essence they are what they mean. A user always has the *Like* option available and can make use of it at any time, in *Posts*, *Comments*, *Pages*, *Profiles* etc. If you like something and you want to express it or want everybody to know about it the *Like* option is available. Concluding this section is a schematic, Figure 1, demonstrating the Facebook platform workflow depicting essentially the use of the *Posts*, *Comments* and *Likes* features.

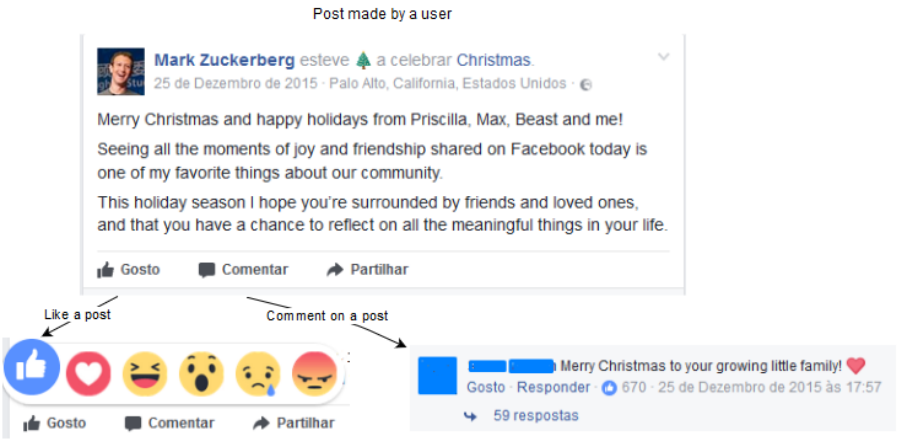

Fig. 1: A user makes a post exposing his opinion to others, allowing them to like and/or comment. The opinion of a commenting user also gets exposed.

Twitter

It's an online news and social networking service where users post and interact through messages, known as *tweets*, restricted to 140 characters. The Twitter social website components, despite completely different in terms of look and feel, allows users to plunge in a similar social experience as Facebook. With an analogy to the Facebook description in the section before, Twitter also permits the user to make *Posts*, known as *tweets*, to react to them using *Likes* and to make *Comments*. A limited yet powerful restriction Twitter has comes in the constraint on characters per *tweet* which must sum at maximum to 140 characters. Such limitation forces the user to be more concise on what he says avoiding extensive text. This is useful because it enables text analysis techniques to be more effective as the content of the *Post* goes straight to the point. Twitter, unlike Facebook, forces all of the user profile to be public enabling anyone to follow it. With Facebook the privacy restrictions make it more difficult to apply data mining techniques as the user is always in control of his profile. Many studies were developed on Twitter, some of which are focused on text analysis techniques applied on user posts that provided useful hints and teachings that will be covered in our final solution.

2.2 Algorithms

In this section a few algorithms considered relevant for the development of this thesis are presented, briefly explaining the most important features and exposing it's importance for the work at hand.

RAKE - Rapid Automatic Keyword Extraction

Extracting keywords is one of the most important tasks when working with text. Readers benefit from keywords because they can judge more quickly whether the text is worth reading, Websites gain from it because they can group similar texts by it's contents, and algorithm programmers profit from it as the text dimensionality gets reduced to the core leaving out the most important features. Keywords are applied frequently to improve the functionality of information retrieval (IR) systems. For example, for search mechanisms based on documents and their content that make use of keywords in order to find similar papers. The case of Google is the most notorious[18][19][20][21].
RAKE (Rapid automatic keyword extraction) is a recent algorithm for keyword extraction released in 2015[23] and is based on the fact that keywords frequently contain multiple words but rarely contain standard punctuation or stop words or others with minimal lexical meaning. For the sake of the explanation of RAKE we will consider a text quote from a Facebook post by Mark Zuckerberg:

Tonight, a Brazilian judge blocked WhatsApp for more than 100 million people who rely on it in her country. We are working hard to get this block reversed. Until then, Facebook Messenger is still active and you can use it to communicate instead. This is a sad day for Brazil. Until today, Brazil has been an ally in creating an open Internet. Brazilians have always been among the most passionate in sharing their voice online. I am stunned that our efforts to protect people's data would result in such an extreme decision by a single judge to punish every person in Brazil who uses WhatsApp. We hope the Brazilian courts quickly reverse course. If you're Brazilian, please make your voice heard.

Rake receives as input a list of stop words, a set of phrase delimiters and a set of word delimiters:

- Stop words: *of, is, to, the, this, and, me, that, a* etc...
- Phrase delimiters: *, ; . - ? !*
- Word delimiters: for example *Wi-Fi* has the hyphen (-) as word delimiter; the result would be *Wi* and *Fi* properly split.

Upon starting keyword extraction on a document, it parses it's content into a set of candidate keywords. First the document set is split into an array of words by the specified word delimiters and after it's split into sequences of contiguous words at phrase delimiters and stop word positions. Words within a sequence are assigned the same position in the text and together are considered as a candidate keyword:

Tonight - Brazilian judge blocked Whatsapp - 100 million people - rely - country - working hard - block reversed - Facebook Messenger - active - communicate instead - sad day – Brazil - Brazil - ally - creating – open internet - Brazilians - always - most passionate - sharing - voice online - stunned - efforts protect peoples data - result extreme decision single judge - punish every person - Brazil

- WhatsApp - hope - Brazilian courts quickly reverse course - Brazilian - please make - voice heard.

The candidate keywords in the order they are parsed from Mark Zuckerbergs post are as above. All stop words, word delimiters and phrase delimiters disappeared leaving only meaningful words and compositions. The next step on the RAKE algorithm procedure is to create a graph of co-occurrences based on the candidate set that was extracted, and assign to each one a score defined as the sum of its member word scores. Many evaluation metrics can be used for calculating word scores (like word frequency *feq(w)*, word degree *deg(w)*, and the ratio of degree to frequency *deg(w)/feq(w)*) but for the purpose of this explanation we will stick to the sum of co-occurrences. For presentation and readability purposes not all extracted keywords will be in Table 1, nevertheless all words are considered for the final result of the algorithm.
Considering the biggest number cluster, involving the words *brazillian, judge, blocked* and *WhatsApp*, all four are extracted together as there are no stop words nor phrase or word delimiters in between words. The word *brazillian* occurs three times in the whole post and occurs once for each word in the sentence.
The case of adjoining keywords can be problematic because RAKE splits candidate keywords by stop words which means extracted keywords do not contain interior stop words therefore expressions like *axis–of–evil* would be senseless. To find these adjoining keywords RAKE looks for pairs of keywords that adjoin one another at least twice in the same document in the same order. A new candidate keyword is then created as a combination of those keywords and their interior stop words then the score is calculated as the sum of its members keyword scores. Finally, after the candidate words are scored, the top T scoring candidates (T is computed as one third of the number of words in the graph [22]) are selected as representative keywords for the document (considering the total computed graph):

- Number of words = 54
- T = $\lceil 54 \div 3 \rceil = 18$
- Document keywords = [*brazilian, brazil, judge, WhatsApp, voice, tonight, blocked, million, people, rely,working, hard, facebook, messenger, active, sad, day, internet*]

The RAKE algorithm achieves higher precision and similar recall in comparison to existing techniques. In contrast to methods that depend on natural language processing techniques to achieve results, RAKE takes a simple set of inputs and automatically extracts keywords in a single pass, making it suitable for a wide range of documents and collections. It's simplicity and efficiency discards the need for high computing power and unlimited resources. As a final reminder, keep in mind that the usage of a dictionary of words, that include synonyms and others, might increase the success and efficiency of RAKE. For example in this case the words *Brazilian* and *Brazilians* are considered in separate but they could very well be merged into a single word saving computational resources in the process.

Table 1: Sum of co-ocurrences for Mark Zuckenberg's post

	tonight	brazilian	judge	blocked	WhatsApp	100	milion	people	rely	Facebook	Messenger	active	communicate	instead	sad	day	Brazil	ally	creating	open	internet	Brazilians	please	make	heard
tonight	1																								
brazilian		3	1	1	1																				
judge		1	2	1	1																				
blocked		1	1	1	1																				
WhatsApp		1	1	1	2																				
100						1	1																		
million						1	1																		
people						1	1	1																	
rely									1																
Facebook										1	1														
Messenger										1	1														
active												1													
communicate													1	1											
instead													1	1											
sad															1	1									
day															1	1									
Brazil																	3								
ally																		1							
creating																			1						
open																				1	1				
internet																				1	1				
brazilians																						1			
please																							1	1	
make																							1	1	
heard																									1

SVM - Support Vector Machines

The SVM is a state of the art supervised learning algorithm developed for pattern classification that has grown in popularity in recent times. The algorithm was proposed by Vladimir Vapnik in 1970[24]. Although the SVM can be applied to various optimization problems such has regression, the classic problem is that of data classification. Figure 2 illustrates the result of applying the SVM algorithm to a group of data points. In the example, the data points are

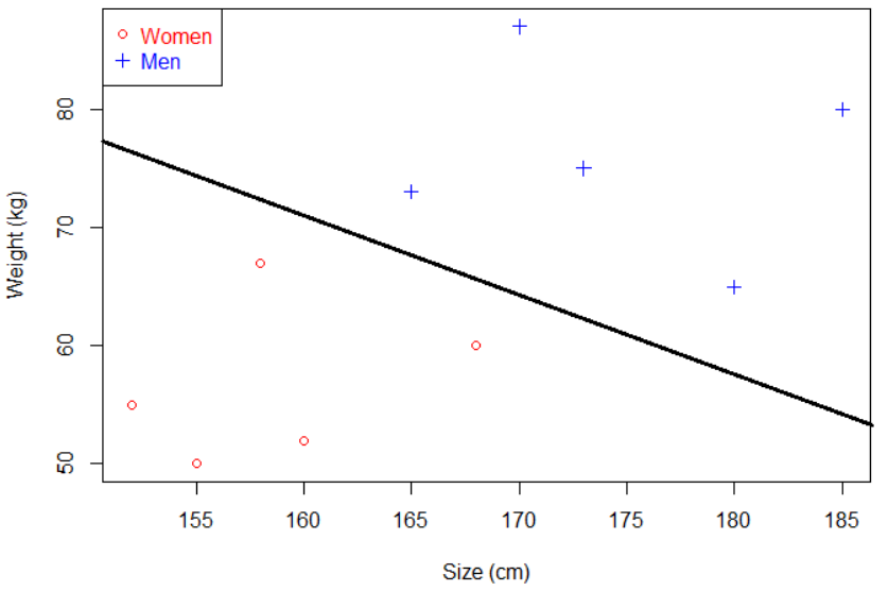

Fig. 2: SVM hyperplane that classifies data as being man or women according to weight and height/size

identified as being men or women, and the goal is to find a hyper-plane that separates each group of data points correctly by a maximal margin. Figure 2 only depicts the 2-dimensional case where the data points, by colors of blue and red, are linearly separable. The definition of hyperplane is therefore subject to the data dimensions:

- one dimension: an hyperplane is known as a point.
- two dimensions: it is a line.
- three dimensions: a plane.
- more than three dimensions: referred to as an hyperplane.

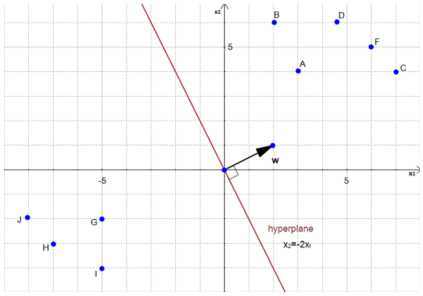

Fig. 3: SVM - Hyperplane with b=0 and the corresponding normal vector $\boldsymbol{w}$

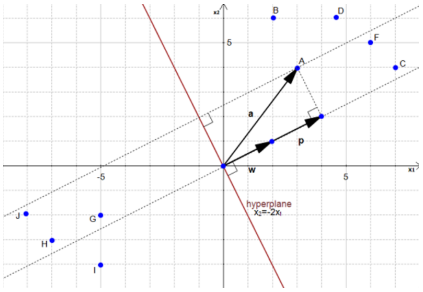

Fig. 4: SVM - Projection of vector $\boldsymbol{a}$ onto the normal $\boldsymbol{w}$ resulting in $\boldsymbol{p}$

The optimal separating hyperplane is the one that is further away from the data points of each category/class, i.e the on which has the highest margin towards each data cluster.

How is the margin calculated?

Considering the case at hand, the hyperplane equation is the classical line equation:

$$y = ax + b$$

or

$$y - ax - b = 0$$

Since more than 2 dimensions can be addressed by the SVM the equation is normally $w^T x = 0$ which translates to, in vector notation:

$$w = \begin{bmatrix} -b \\ -a \\ 1 \end{bmatrix}, x = \begin{bmatrix} 1 \\ x \\ y \end{bmatrix}$$

Following this reasoning we will reach the conclusion that $w^T x = y - ax - b$. In Figure 3 is an hyperplane which separates 2 groups of data.
Considering $w0 = 0$ the hyperplane $x_2 = -2x_1$ or $x_2 + 2x_1 = 0$ will cross the origin of the referential. Translating to vector notation

$$w = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

The distance to the point $A(3, 4)$ from the line would be the distance between A and its projection onto the plane, this would result in vector $\boldsymbol{p}$ which is the projection of A against the normal vector of the line, denoted as $\boldsymbol{w}$. The distance and result of this projection would be vector $\boldsymbol{p}$ (Figure 4). Assuming the $\boldsymbol{u}$ as the direction of $\boldsymbol{w}$,the normal to the line, as

$$\boldsymbol{u} = \begin{bmatrix} 2/\sqrt{5} \\ 1/\sqrt{5} \end{bmatrix}$$

the magnitude or length of vector $\boldsymbol{p}$ is $\|\boldsymbol{p}\|$ and is given by the formula $(\boldsymbol{u} \cdot \boldsymbol{a})\boldsymbol{u}$ where $\boldsymbol{a}$ is a vector coming from the origin to point A.
Concluding this example the value for $\|\boldsymbol{p}\|$ is $\sqrt{4^2 + 2^2} = 2\sqrt{5}$ and the margin to point A is computed as $2\|\boldsymbol{p}\| = 4\sqrt{5}$.
Recalling that the final objective of the SVM is to maximize the margin between the hyperplane and each data cluster, amongst all possible hyperplanes meeting the constraints, the chosen one will be the one with the smallest norm $\|\boldsymbol{w}\|$ ($\boldsymbol{w}$ is refers to the properties of the hyperplane) equivalent to the biggest margin.
Therefore, the search for the optimal margin is an optimization problem focused on minimizing in (w, b) the $\|\boldsymbol{w}\|$, subject to $y_i(w \cdot x_i + b) \geqslant 0$ for any $i = 1, 2, ..., n$.
Solving the equation will return the smallest value for $\|\boldsymbol{w}\|$.
For a more abstract example of the SVM we could imagine plotting on a 2D space a few data points where each one would represent a person with certain characteristics that would fit into a specific category known beforehand. The SVM, after finding the desired line, would be able to receive an unknown data point representing a person and assign him or her to one of the two categories in either side of the hyperplane.
The example described in this section only covers the 2 class categorization, however the SVMs are also applicable for multi-class scenarios which have much more to offer.

KNN - K-Nearest Neighbors

The KNN is a non-parametric method used for classification and regression that has been used for estimation and pattern recognition since 1970[25]. It can be classified as a supervised learning algorithm as it makes use of previous known data in order to classify new information. For this work the KNN will be used as a mean for classification (not regression) on which the output of the classifier will be a class membership.
To understand the K-Nearest-Neighbors classification algorithm we will use a simple example on which we will attempt to classify an new data point using the *neighbors* perspective (Figure 5).

The first step to build a KNN classifier is to train it with data which the category is already known. For this example represented in Figure 5 the classifier was trained using the data represented by the green squares and blue hexagons. The goal is to accurately classify new incoming data, like the red star, as a green square or as a blue hexagon. But how exactly is this classification done?
As the algorithm name indicates the classification is done through the nearest neighbors of the new data point. For the example at hand the red star is plotted side by side with the other data points for which the class is already known. To classify the new data point the KNN considers a group of points, more specifically K points which are the K points nearest to the one to be classified as a green square or blue hexagon. In order to assess the selection of the nearest K points the algorithm considers one of the following similarity measures:

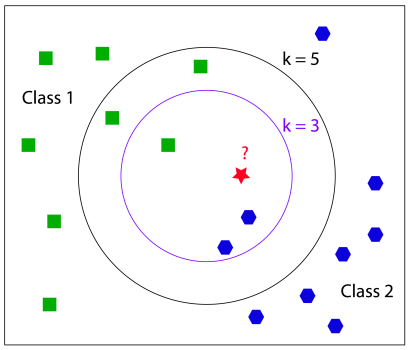

Fig. 5: KNN algorithm for classification

– Euclidean distance
– Manhattan distance
– Minkowski distance

The Euclidean distance formula states that the distance between 2 points is equal to the length of the line segment connecting them. It's formula is as below:

$$d(\boldsymbol{p}, \boldsymbol{q}) = \sqrt{\sum_{i=1}^{n} (\boldsymbol{q_i} - \boldsymbol{p_i})^2}$$

The value of n is dependent of the number of space-dimensions. In this case $n = 2$, for the x and y dimensions.
The Manhattan function computes the distance that would be traveled to get from one data point to the other if a grid-like path is followed. The distance between two items is the sum of the differences of their corresponding components for x and y in case of a two-dimensional space. The computation is done through the formula:

$$d = \sum_{i=1}^{n} |x_i - y_i|$$

Like before the value of n is variable according to the number of dimensions. Finally the Minkowski distance, a generalization of the Euclidean and Manhattan distance, defines the distance between two points in a normed vector space and is calculated by:

$$d(\boldsymbol{x}, \boldsymbol{y}) = (\sum_{i=0}^{n-1} |x_i - y_i|^p)^{1/p}$$

if p = 1 the distance given by the Minkowski function will be equivalent to the Manhattan distance and if p = 2 it would result in the Euclidean function. One important aspect to keep in mind for the three measures above is that, if the variables are not continuous none of the three distance functions could be used.

Instead the Hamming distance should be addressed, which description is not covered in this paper.
Having a generic understanding of the distance measures we are now able to classify our red star data point. For that we will need to consider K neighbors nearest to the point. We will consider K with the value of 3 and with the value of 5. Considering the results in Table 2, for 3 neighbors the red star is classified as a blue hexagon and for K=5 it's reversed to a green square.

Table 2: *K-neighbors* and respective class assignment for the red star

K	Class
1	Blue
2	Blue
3	Blue
4	Blue or Green
5	Green

The best choice of K depends upon the data; generally, larger values of K reduce the effect of noise on the classification[26].
Therefore the classification using the KNN algorithm is always subjective to the number of neighbors that are considered when classifying, nevertheless, similarly to the SVMs the KNN is a state of the art algorithm that has proven very effective and with considerable success rates upon classification.

LASSO - Least Absolute Shrinkage and Selection Operator

LASSO is a regression analysis method first introduced by Robert Tibshirani in 1996 based on Leo Breimans Nonnegative Garrote[27][28].
LASSO performs both variable selection and regularization in order to enhance the prediction accuracy and reduce the dimensionality of features by removing the *noise*. The algorithm performs a penalization on extra features by penalizing the absolute size of the regression coefficients. Imagine a set of features x_1, x_2, x_3, x_4 and the corresponding regression coefficients w_1, w_2, w_3, w_4. The result of mapping the features and corresponding coefficients into a regression model would be:

$$y = w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + b$$

where b is the bias.

The penalizing method for LASSO regression occurs in the coefficient adjustment which can be set to zero.
Consider the following scenario: Upon analyzing the characteristics of a indefinite number of Basketball players 4 features stood out as important (but not on the same level) such as: *height, physical ability, speed and agility* that we will denote as x_1, x_2, x_3, x_4. After analyzing the dataset of players we came to

the conclusion that the most successful players were the highest followed by the most agile and with great physique. Mapping this hypothetical results into a regression model would mean that in order to reduce the noise of our model the feature with less weight would be set to zero by simply adjusting the corresponding weight to zero ($w = 0$) and recalibrating the remaining weights.
This would mean that a player can still be accurately classified without the *speed* capability as it's not *that relevant*.
The goal of LASSO is to minimize:

$$\sum_{i=1}^{n}(y_i - \sum_{j} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p}|\beta_j|$$

which is the same as minimizing the sum of squares with constraint to $\sum|\beta_j| \leqslant s$. Some of the β's are shrunk to exactly zero, resulting in a regression model that is easier to interpret. The tuning parameter λ controls the amount of *shrinkage*:

- if $\lambda = 0$ no parameters are eliminated
- as λ increases more and more coefficients will be set to zero and eliminated

LASSO is one of the most used methods for feature selection together with the Ridge regression, nevertheless performing quadratic computations is heavy, therefore a lot of time and computational power may be required depending on the dimensions of the dataset.

2.3 State of the Art

Social networking on the web has grown dramatically over the last decade. As more and more people live their lives online, social media sites have become a huge source of information. Recently, online communities have been exploited as a source of capital, especially for career promotion and marketing campaigns.
The sharing of ideas and opinions have also extended to political discussions where users express their perspective relative to each political party through their available online pages. This political oriented sharing enabled data mining experts to build a prediction model for political orientation of users that has proven to be effective in recent campaigns such as the France elections of 2017 and the USA presidential elections won by Donald Trump.
This section describes a few works made on online personality assessment using Facebook Likes and concludes with the analysis of an experience that covered the usage of Facebook for prediction of political orientation of it's users.

Personality prediction using Facebook Likes

The understanding of people's personality traits is an essential component of social living. Everyday we assess a person's personality and make judgments about him/her without even noticing[29][30]. This evaluations made about others can sometimes define the success or failure of our decisions in relation to

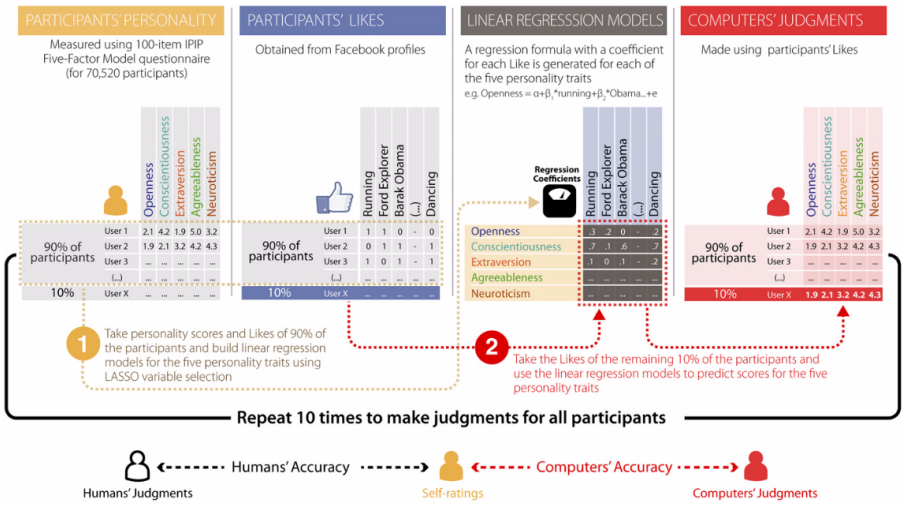

Fig. 6: Methodology used to obtain computer-based judgments and estimate the self-other agreement[12].

them so it's crucial to make an accurate judgment in order to derive the best possible outcome[31][32][33].

Recently, scholars conducted a study with the objective of finding if computer-based personality judgments using Facebook *Likes* functionality could produce equal or better success rates than judgments made by humans alone[12]. The result was surprising, the accuracy of computer-based judgments considerably exceeded the success of human social-cognitive skills.

The study was conducted based on Facebook *Likes* obtained by around 70,000 participants. The first step used by the researchers was to use LASSO (Least Absolute Shrinkage and Selection Operator) linear regression[27] with 10-fold cross-validations so that judgments for each participant were made using models developed on a different subsample of participants and their *Likes*, thus avoiding over fitting.

The human personality judgments were obtained from the participants Facebook friends who were asked to describe a given participant using a 10-item version of the IPIP personality measure[34]. The method used for this process confirmed by it's success is the most important teaching extracted for this thesis and is depicted in Figure 6.

Participants and their *Likes* are represented as a matrix where entries are set to one if there is a correspondence between the user and the topic of the *Like* and to zero otherwise - second panel. Next, the matrix is used to fit LASSO linear regression model, one for each Big Five personality trait - third panel (note that the weights given to each *Like* subject is different). A ten-fold cross validation is applied to avoid over fitting:

– The sample is randomly divided into 10 equal-sized subsets where 9 out of 10 are used to train the model
– The trained model is then applied to the remaining subset to predict the personality score for each participant

The above procedure is repeated 10 times to predict the personality for the entire sample.
The key point of this procedure for our work, aside from its success, is the mapping of the user *Likes* matrix to the LASSO regression model enabling a more direct control over the model and its features.

Further studies by common authors aimed at the Facebook platform proved that easily accessible digital records of behavior - *Likes* - can be used to automatically and accurately predict a range of highly sensitive personal attributes including: sexual orientation, ethnicity, religious and political views, personal traits, intelligence, happiness, use of addictive substances, parental separation, age, and gender[11]. This research was based on a dataset of around 58,000 volunteers who provided access to their Facebook *Likes*.
The design/implementation of this study is represented in the figure. Similarly to the previous explained work, the users and their *Likes* were represented as a sparse user-*Like* matrix representing associations. The dimensionality of the user matrix was reduced using the SVD (Single Value Decomposition) technique. Numeric variables such as age or intelligence were predicted using a linear regression model, whereas dichotomous variables like gender or sexual orientation were predicted using logistic regression. In both cases, analogous to the previous, 10-fold cross validation was applied to avoid over fitting (Figure 7).

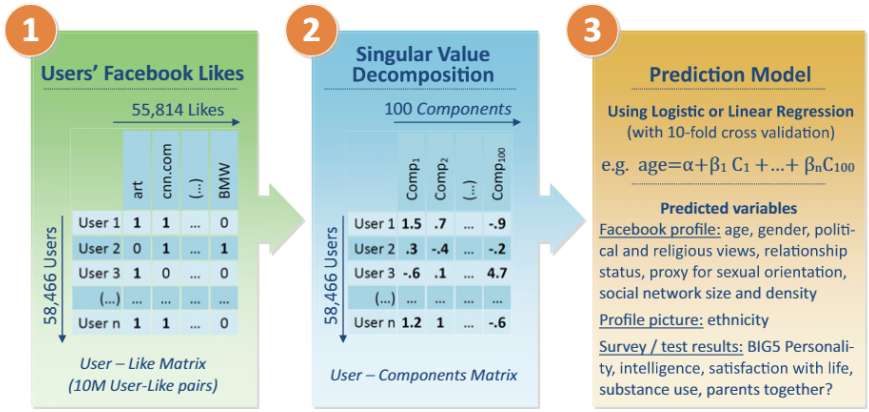

Fig. 7: Methodology used to obtain the prediction model for the user personality throughout his *Likes*[11].

Both works were proven successful when it comes to predict the user personality by his/her Facebook *Likes*. The methodology used in both studies was pretty similar, the first used the LASSO regression to generate a accurate model while the other made use of the Single Value Decomposition (SVD) technique to reduce the matrix dimensionality and afterwards mapping it to a prediction model using Logistic and Linear regression.
The methodologies used in the previous works are very important and will be further referenced in the ***Solution Proposal*** chapter.

Using Facebook to predict Political Orientation of users

Nowadays every political party and leader maintains an account on Facebook, Twitter, or any other social platform, where they publish recent updates according to their agenda. Recently many politicians and their strategists use social networks as an effective platform to influence the agendas of professional journalists and appeal to their supporters. They also attempt to employ them to directly communicate with their electorate and build community support. In order to communicate and appeal directly, correctly and assertively politicians need to identify the potential electorate among different users on the network site. This paper[36] aims at finding and assigning a user to one of three main categories:

- Left wing supporter
- Center wing supporter
- Right wing supporter

Using text mining and categorization techniques researchers weren't only able to classify each user but also to identify *swing voters* by analyzing comment's and posts made by users on political party pages as well as in each user personal profile. A brief explanation is described next.

Data Collection

The dataset collected for this experiment was retrieved from 2 places each with relevant meaning. The first dataset was taken from Facebook pages of political parties and it consisted mostly on text content from posts. This first dataset is important because beforehand we are able to know for sure from which political party it comes from, thus enabling an association between each party/wing and the corresponding texts. The second dataset was assembled from personal Facebook pages of volunteers (around 450) who agreed to participate in the experiment. This group of volunteers filled a questionnaire upon agreeing to the experiment so that their political tastes were known to the researchers.

Setup and Experiment

The first step taken was the categorization of the textual input of the dataset. Each text in a set of labeled example texts (from political parties pages) is represented as a numerical vector reflecting the frequencies in the text of each feature in a specified feature set. Table 3 illustrates an example of a vector.

Table 3: Numerical Vector representing word frequencies

Word	Frequency
they	8
them	3
refugee	2
luck	0
strong	1
...	...

Afterwards a machine learning algorithm is used to learn a classifier that best distinguishes the training examples in different classes. In the end these classifiers can be used to classify new texts. The effectiveness of this method can be measured by applying a learned classifier to labeled test texts for which the correct answer is given. For example, the use of k-fold cross validation.
The training set is divided into k equal parts whereas k-1 sets are used for training and the remaining one is used for testing. This is then repeated k times with a different test set each time. In this context the following experiments were conducted:

- For each corpora (text group), coming from political party pages or user profile pages, tenfold (K=10) cross validation was performed to determine the accuracy with which a political preference classifier can be trained for a given corpus.
- A classifier is trained on training data in political party pages and it's effectiveness is tested against personal Facebook pages.
- A learned two-class(left/right wing) classifier is used to determine whether self-identified centrist voters(information coming from the questionnaire) are closer to the left or right. A three-class (left/center/right) classifier is used to determine the accuracy with which left, center or right learned texts can be distinguished.
- A final tenfold cross-validation experiment is conducted to determine the accuracy with which a classifier can identify *swing voters*.

The results of this experiment proved that political views from each author on his/her Facebook page can be recognized with very high accuracy, nearly 90%, when the classifier is trained on other personal Facebook pages from the same corpus. Classifiers trained on a political genre can be used to effectively classify non-political personal Facebook pages reducing the need for manually annotating personal pages for training data.

One of the most important discoveries made in this work was that each political wing has a set of terms that best describe them, for example right wing parties use certain words (*God, Rabbi and Amen*) that left oriented parties don't and vice-versa.
Although this research was conducted only on Israeli Facebook users its findings were extremely helpful for the progress of this thesis.

2.4 Conclusions

Throughout this entire chapter some studies and techniques were described, some which will be used and exploited for the solution this work proposes.
In the ***State of the Art*** section a few works highly connected to the one proposed were examined and their relevant and most important features were described. The next section describes the proposed solution and architecture of the algorithm to be developed starting with a general description and then detailing each module that composes it.

3 Solution Proposal

In this section the project architecture and structure is described, starting with an introduction to the solution schema divided in two phases: the first which considers the gathering and physical storage of data, and the second which implies the development of two machine learning classifiers, one for text and another for Facebook *Likes*. At the end both classifiers will be used for the categorization of all Facebook users whose information is stored in the database.

3.1 Architecture Overview

This project's architecture (Figure 8) is partitioned in two main sections each with a different objective which highly relate to Data Mining related processes. The sections are as follows:

- Data Pre-processing
- Data Classification

The first phase (**Data Pre-processing**) objective is mainly data collection and organization in a relational database system that will then be used to train two classifiers. At this stage all the data requested from the Facebook **Graph API** which considers user information, *Posts*, *Likes*, *Comments*, as well as political party pages data, is to be parsed and properly stored in order to ease the complexity of the next phase. The second phase (**Data Classification**) cares about the data evaluation and analysis where the algorithms described in the *State of the Art* section will be considered. It's at this phase that the two classifiers, for text and *Likes*, will be developed, trained and put to the test. For both training and testing the classifiers will require the data that is hosted in the database, acquired in the **Data Pre-processing** stage.

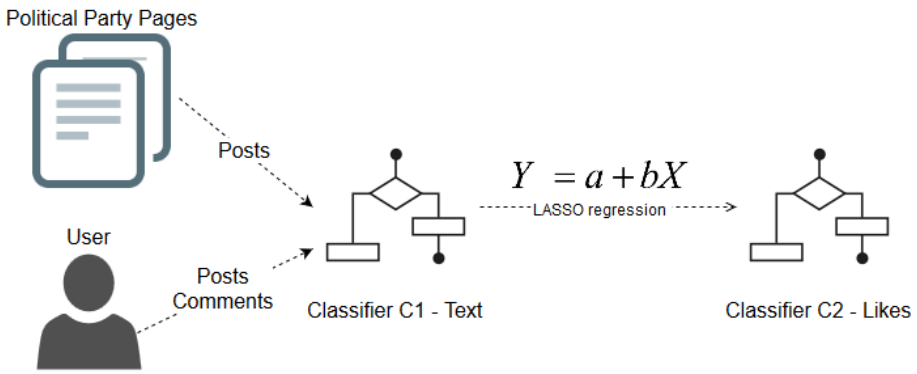

Fig. 8: High level solution description

3.2 Phase 1 - Data Pre-processing

This phase main objective, as it was described before, is about data collection and organization. When preparing a Data Mining related work this is always the first step to take. Large data mining applications make use of a Data Warehouse or Data Mart to store all collected information for further use. In this case a relational database will be used which will probably not achieve the dimensions of a DW. In the first section of this thesis proposal the reader was introduced to the Facebook social network main features and components, which would later be quoted in each one of the analyzed works in the *State of the Art* section. Well, in this first phase all those features are assembled and stored in the database accordingly. Figure 9 attempts to provide a visual representation of this phase.

In order to acquire the necessary information the algorithm must make use of Facebook's official API, named *Graph API*, which is publicly accessible. The API tolerates only the use REST oriented communications. The information requested from the API comes from two different sources:

- The first source are the political party pages *(PPP)*, which are public and therefore the Facebook *Graph API* imposes no restrictions for the data extraction, it's only needed to know which party pages are going to be targeted for extraction.
  The data to be extracted from the Political Pages will be the posts made by the politicians who represent such page, the comments made on those posts and finally the information about users that commented or posted in that specific page.
- The second source is the profile page for each Facebook user as well as his major online activity information coming from the Feed or other pages that may not be of political parties. In order to be able to access the user profile and activity information through the Graph API it's required that the user gives permission to do so. This is required due to the user profile privacy

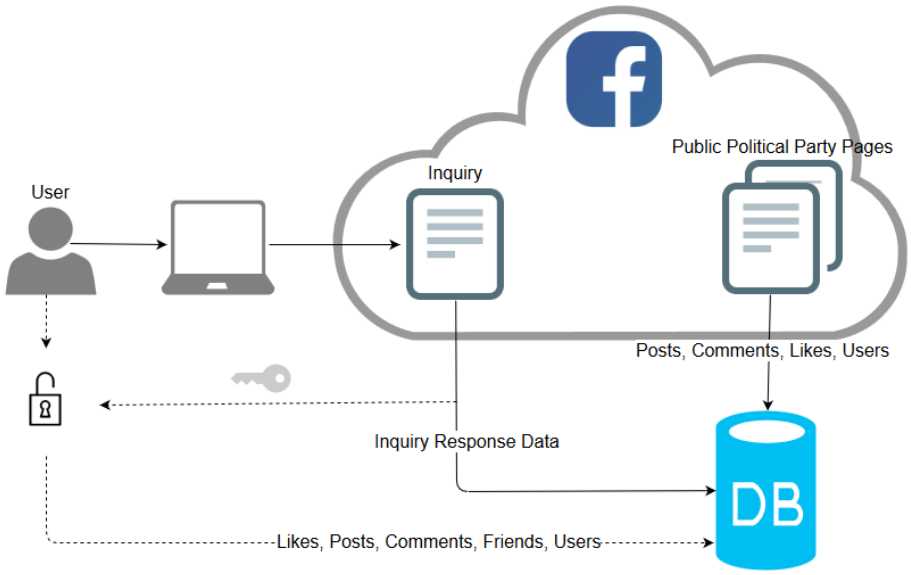

Fig. 9: Phase 1 - Data pre-processing

restrictions which don't apply if the user profile is completely public. For these reasons it's required that the user answers a questionnaire that will ask the him permission of access to his information and activity.

The user questionnaire is the most important enabler for this project because if the user doesn't authorize his information to be analyzed through inquiry we are unable to move forward with that user and respective friends.
At the end, this information altogether will be stored in a structured database so that it can be parsed in the next phase.

3.3 Phase 2 - Data Classification

The Data classification phase aims at making sense of all the extracted information, meaning the prediction of the political orientation of users throughout two classifiers of text and likes by making use of some data mining and machine learning algorithms. All the data previously stored in the database will be parsed, normalized and at the end, the resulting meaningful information will be stored as well. The trouble to understand this phase can be a bit overwhelming, therefore the explanation was divided in 4 illustrations that sequentially create the logic for the solution.
The first step (Figure 10) is to train a classifier based on the text from Political Party Pages that after trained will be able to classify new texts and assign each to each class. The text that will input this classifier C1 is filtered using RAKE in order to be trained on keywords and meaningful text only. This classifier will

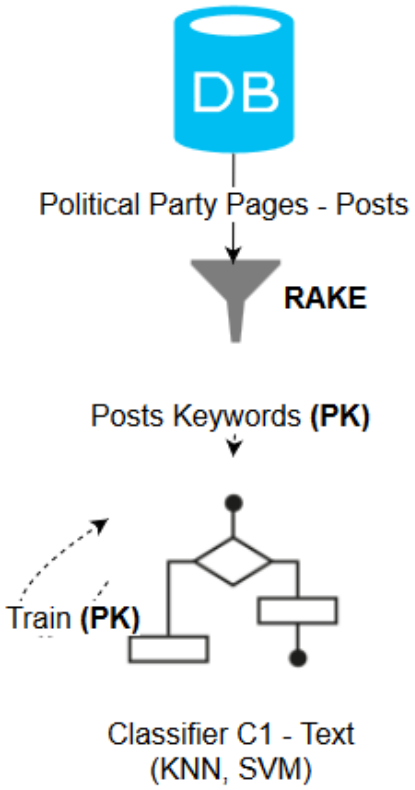

Fig. 10: Step 1 - Train a classifier C1 based on text from *PPP* Posts

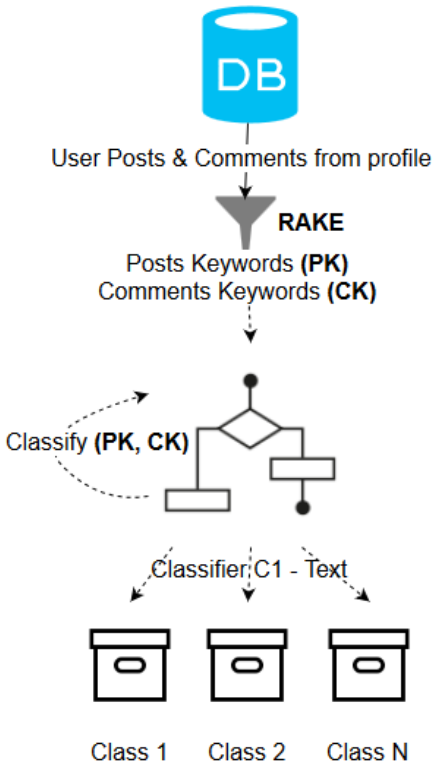

Fig. 11: Step 2 - Classify each user using C1 through user profile posts and comments

be based on the KNN algorithm and the SVM classification technique. At the end of this step classifier C1 will be capable of receiving new keywords and texts and based on it's contents assign those text to a specific class which will be at this scope a political party representative of the Portuguese elections.

After C1 is trained, in step two (Figure 11), it will be used to classify *Posts* and *Comments* from each user and assign each one to a certain class based on his texts filtered beforehand using RAKE. As each user is assigned to a class the *assignment* information or *links*, will be stored in the database so that the user and his corresponding class are not lost afterwards.

The 3rd step (Figure 12) is where the Facebook *Likes* are considered. At this point we have successfully classified users into specific classes, using a text classifier C1, and have stored those associations in the database. Now, for each class and corresponding users the algorithm will extract the *Likes* of all users and a user–*Likes* matrix will be built. From this matrix we will extract all the

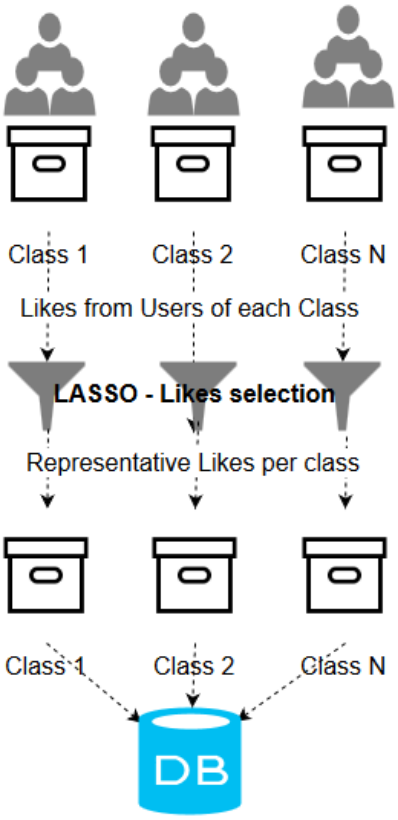

Fig. 12: Step 3 - Extract user likes from each class and select the most representative per class using LASSO

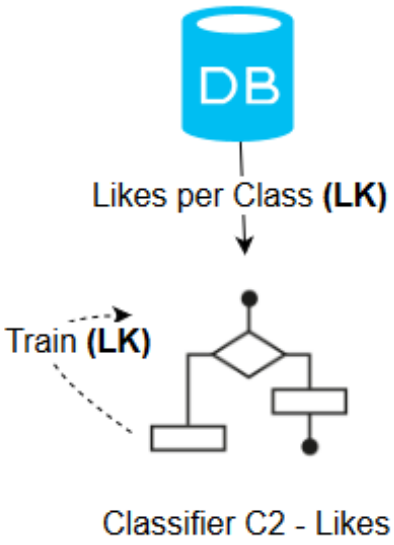

Fig. 13: Step 4 - Train a classifier C2 based on the Likes extracted from Step 3

Likes, weight them according to their frequency in the matrix and finally, build a LASSO linear regression.
This equation will have all likes and the correspondent weight associated altogether. After this transformation is complete, the most relevant likes will be extracted by adjusting the regression model to a value of choice disposing of all the likes whose weight is not significant for the adjustment. The *Likes* that resulted from the regression will be stored in the database as representatives of the class they came from.
Finally, Step 4 (Figure 13) would be to train a new classifier C2 based on the Likes of each class discovered by the LASSO regression. This new classifier C2 should be capable of assigning new unknown users to a class based on their Likes only. Notice that this new classifier is created indirectly from classifier C1 therefore its accuracy might drop in comparison.
At the end of this process we would possess two classifiers in total:

– Classifier C1 for text (*Posts* and *Comments*)
– Classifier C2 for *Likes*

3.4 Integration and Execution

After the process of training the classifiers we will have stored in the database the information of the users who answered the questionnaire, more precisely the class they belong to which can be for example Left, Center-Left or Center-Right wing and even the specific party in that class, in regard to the political parties represented in the Portuguese parliament in the present[39]. The most notorious political parties and respective classes today are:

– **Left wing**: *Portuguese Communist Party* (PCP), *Left Bloc* (BE)
– **Center Left wing**: *Socialist Party* (PS)
– **Center Right wing**: *Social Democratic Party* (PSD), *CDS People's Party* (CDS)

The next step, after having the two classifiers (C1 and C2) properly trained, will be to extend the classification to the Friends of the users who answered the inquiry and if possible to other users whose public information is available. This will imply a merging of C1 and C2 classifiers into a single well integrated algorithm.

3.5 Expected Results

It is expected that the number of users whose information is stored in the database grows way beyond the number of users that solely answered the questionnaire. By having this much users we hope that the final goal of this thesis, which is to predict the election results, can be accomplished with high accuracy.

4 Evaluation Methodology

In order to measure the performance of the solution it was necessary to resort to the main papers which were the pioneers in the mining of social networks. We will consider the following indicators:

– Number of users reached through the questionnaire.
– Number of friends of a user that has answered the questionnaire (considering that the average number of friends of an adult Facebook user is around 400 [38])
– Amount of keywords stored in the database
– Number of Likes stored in the database
– Number of political party pages gathered
– Number of single users withdrawn from political pages and other sources

The indicators above should be able to reflect the dimensionality and the proportion the algorithm might achieve. The number of users reached is perhaps the most important has these users will be used not only in the training of the classifier but will also represent the amount of friends that might be processed as well. Also, in the analysis of the political pages, it's expected that all major party pages are analyzed and their contents used for the training of the classifier. The more information gathered from this pages and from the user that has answered the inquiry, the more accurate may be the trained classifier.
In terms of time related performance this algorithm falls short. Since it will be highly related to the machine learning area it's response is not direct to the user as a website is expected to be. Therefore, in this case the algorithm scaling and time consumption evaluations will not apply. Finally the most important assessment will be whether or not the algorithm is capable of grasping with considerable success the result of the Portuguese elections.

5 Calendar

This section introduces a schema describing the time mapping for the thesis development. In Figure 14, this schema is presented whereas the most important tasks, their meaning and respective deadline are registered as a *Gantt* chart.

6 Conclusions

In this document we propose a solution with the objective of predicting the political orientation of a Facebook user through his *Posts*, *Comments* and *Likes*. We make use of two classification algorithms based on Support Vector Machines, K-Nearest Neighbors, the recently developed text extractor RAKE and the LASSO regression technique for feature selection. A few works representative of the *State of the Art* were introduced and explored as motivational topics for this solution. We expect this algorithm to achieve better results than previous attempts by implementing a new, innovative and refreshed solution that may be a useful mark for the future of politics.

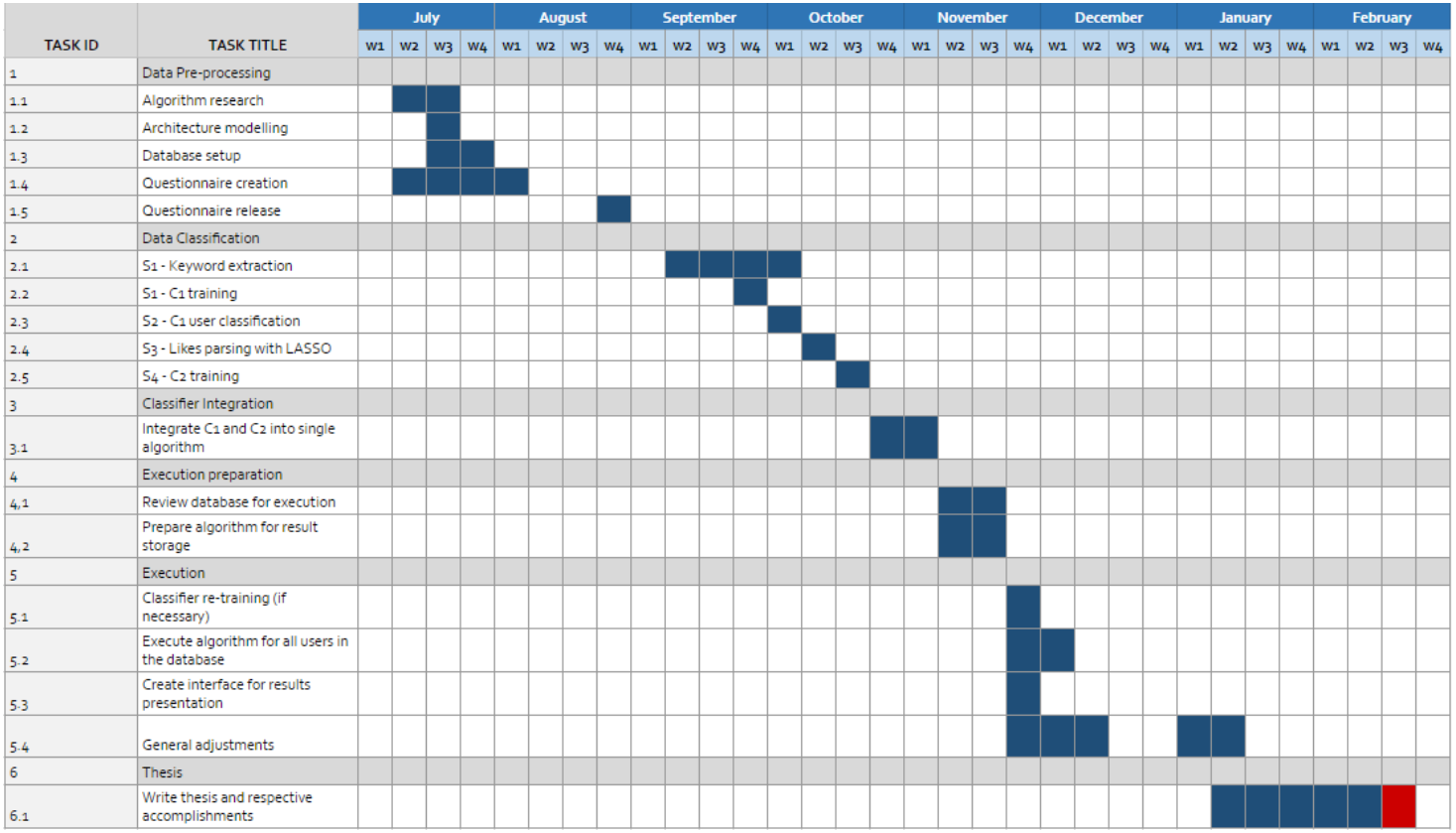

Fig. 14: Calendar

References

1. Ellison, N. B., Steinfield, C., & Lampe, C. (2007). The benefits of Facebook friends: Social capital and college students use of online social network sites. Journal of ComputerMediated Communication, 12(4), 1143-1168.
2. Burke, M., Kraut, R., & Marlow, C. (2011, May). Social capital on Facebook: Differentiating uses and users. In Proceedings of the SIGCHI conference on human factors in computing systems (pp. 571-580). ACM.
3. Steinfield, C., Ellison, N. B., & Lampe, C. (2008). Social capital, self-esteem, and use of online social network sites: A longitudinal analysis. Journal of Applied Developmental Psychology, 29(6), 434-445.
4. Pesonen, J. (2011, January). Tourism marketing in facebook: Comparing rural tourism SMEs and larger tourism companies in Finland. In ENTER (pp. 537-546).
5. Weinberg, T. (2009). The new community rules: Marketing on the social web. " O'Reilly Media, Inc.".
6. Stieglitz, S., & Dang-Xuan, L. (2013). Social media and political communication: a social media analytics framework. Social Network Analysis and Mining, 3(4), 1277-1291.
7. Williams, C. B., & Gulati, G. J. J. (2013). Social networks in political campaigns: Facebook and the congressional elections of 2006 and 2008. New Media & Society, 15(1), 52-71.
8. Kim, Y. (2011). The contribution of social network sites to exposure to political difference: The relationships among SNSs, online political messaging, and exposure to cross-cutting perspectives. Computers in Human Behavior, 27(2), 971-977.
9. Baek, Y. M. (2015). Political mobilization through social network sites: The mobilizing power of political messages received from SNS friends. Computers in Human Behavior, 44, 12-19.
10. Kreiss, D. (2016). Seizing the moment: The presidential campaigns use of Twitter during the 2012 electoral cycle. New Media & Society, 18(8), 1473-1490.
11. Kosinski, M., Stillwell, D., & Graepel, T. (2013). Private traits and attributes are predictable from digital records of human behavior. Proceedings of the National Academy of Sciences, 110(15), 5802-5805.
12. Youyou, W., Kosinski, M., & Stillwell, D. (2015). Computer-based personality judgments are more accurate than those made by humans. Proceedings of the National Academy of Sciences, 112(4), 1036-1040.
13. Statista (2017). "Global social networks ranked by number of users" - *www.statista.com*
14. Alexa Internet (2016). "Friendster.com Site Info" - *http://www.alexa.com/*
15. Techtree (2008). "Facebook: Largest, Fastest Growing Social Network" - *http://www.techtree.com/*.
16. Carlson, Nicholas (2010). "At Last The Full Story Of How Facebook Was Founded". Business Insider. Axel Springer SE.
17. Phillips, Sarah (2007). "A brief history of Facebook". The Guardian. Guardian Media Group.
18. Jones, S., & Paynter, G. W. (2002). Automatic extraction of document keyphrases for use in digital libraries: evaluation and applications. Journal of the American Society for Information Science and Technology, 53(8), 653-677.
19. Gutwin, C., Paynter, G., Witten, I., Nevill-Manning, C., & Frank, E. (1999). Improving browsing in digital libraries with keyphrase indexes. Decision Support Systems, 27(1), 81-104.

20. Hulth, A. (2004). Combining machine learning and natural language processing for automatic keyword extraction. Department of Computer and Systems Sciences [Institutionen fr Data-och systemvetenskap], Univ..
21. Andrade, M. A., & Valencia, A. (1998). Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. Bioinformatics, 14(7), 600-607.
22. Mihalcea, R., & Tarau, P. (2004, July). TextRank: Bringing order into texts. Association for Computational Linguistics.
23. Rose, S., Engel, D., Cramer, N., & Cowley, W. (2010). Automatic keyword extraction from individual documents. Text Mining, 1-20.
24. Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine learning, 20(3), 273-297.
25. Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician, 46(3), 175-185.
26. Everitt, B. S., Landau, S., Leese, M., & Stahl, D. (2011). Miscellaneous clustering methods. Cluster Analysis, 5th Edition, 215-255.
27. Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society. Series B (Methodological), 267-288.
28. Breiman, L. (1995). Better subset regression using the nonnegative garrote. Technometrics, 37(4), 373-384.
29. Funder, D. C. (2012). Accurate personality judgment. Current Directions in Psychological Science, 21(3), 177-182.
30. Letzring, T. D. (2008). The good judge of personality: Characteristics, behaviors, and observer accuracy. Journal of research in personality, 42(4), 914-932.
31. Funder, D. C., & West, S. G. (1993). Consensus, selfother agreement, and accuracy in personality judgment: An introduction. Journal of personality, 61(4), 457-476.
32. Letzring, T. D., & Human, L. J. (2014). An examination of information quality as a moderator of accurate personality judgment. Journal of personality, 82(5), 440-451.
33. Funder, D. C. (1995). On the accuracy of personality judgment: a realistic approach. Psychological review, 102(4), 652.
34. IPIP, International Personality Item Pool *http://ipip.ori.org/ipip/*
35. Golbeck, J., Robles, C., & Turner, K. (2011, May). Predicting personality with social media. In CHI'11 extended abstracts on human factors in computing systems (pp. 253-262). ACM.
36. David, E., David, E., Zhitomirsky-Geffet, M., Zhitomirsky-Geffet, M., Koppel, M., Koppel, M., ... & Uzan, H. (2016). Utilizing Facebook pages of the political parties to automatically predict the political orientation of Facebook users. Online Information Review, 40(5), 610-623.
37. Internet World Stats - *http://www.internetworldstats.com/europa.htm*
38. Pew Research Center - Facebook statistics - *http://www.pewresearch.org/fact-tank/2014/02/03/6-new-facts-about-facebook/*
39. Portuguese Parliament (2017). *https://www.parlamento.pt/DeputadoGP/Paginas/GruposParlamentaresI.aspx*