University of Pisa

Master's Degree in Artificial Intelligence and Data Engineering

---

Classification and Stance Analysis of tweets regarding Italian politicians

---

Students:
Lorenzo Biondi
Nicola Bicchielli

Professors:
Prof. Francesco Marcelloni
Prof. Pietro Ducange

Academic Year 2020-2021

Contents

Chapter 1

Introduction

In this project we decided to study the replies some Italian politicians received on Twitter throughout the course of 2020. Our goal was not only to understand how we could classify these messages (*tweets*), but also to understand how the general *Public Opinion* towards the politicians changed over in time.

1.1 Classification Steps

In the following paragraphs we present the different steps that we followed to correctly classify the tweets. These steps make up the classical machine learning process that is typically followed in this scenario. First of all, it was necessary to make a collection of the tweets to analyse. To obtain these tweets, we used a Python library called "Twint". This library is capable of obtaining the same results as the official Twitter API but allowed us to download tweets that were written more than one week earlier. Altogether more than 3 million tweets were stored in our dataset, which was made up of not only the tweets themselves but included other fields such as the timestamp, the id of the author and the id of the users to which the tweet was addressed to. The following list describes the politicians we studied which are all well-known in the Italian scenario (in brackets the politician's Twitter username):

- Alberto Cirio, President of Piedmont (*Alberto_Cirio*)
- Antonio Tajani, Vicepresident of Forza Italia (*Antonio_Tajani*)
- Attilio Fontana, President of Lombardy (*FontanaPres*)
- Eugenio Giani, President of Tuscany (*EugenioGiani*)
- Giorgia Meloni, leader of "Fratelli d'Italia" (*GiorgiaMeloni*)
- Giovanni Toti, President of Liguria (*GiovanniToti*)
- Giuseppe Conte, Prime Minister of Italy (*GiuseppeConteIT*)
- Luca Zaia, President of Venento (*zaiapresidente*)
- Lucia Azzolina, Minister of Public Education (*AzzolinaLucia*)
- Luigi Di Maio, leader of " Movimento 5 Stelle" (*luigidimaio*)
- Matteo Renzi, leader of "Italia Viva" (*matteorenzi*)
- Matteo Salvini, leader of "Lega Nord" (*matteosalvinimi*)

- Nicola Zingaretti, President of Lazio and Secretary of "Partito Democratico" (*nzingaretti*)
- Roberto Gualtieri, Minister of Economy and Finance (*gualtierieurope*)
- Roberto Speranza, Minister of Health (*robersperanza*)
- Silvio Berlusconi, President of "Forza Italia" (*berlusconi*)
- Stefano Bonaccini, President of Emilia-Romagna (*sbonaccini*)
- Vincenzo De Luca, President of Campania (*VincenzoDeLuca*)
- Virginia Raggi, Mayor of Rome (*virginiaraggi*)
- Vito Crimi, leader of "Movimento 5 Stelle" (*vitocrimi*)

After having downloaded all these tweets, we cleaned them by removing duplicates and non-Italian tweets. We then selected and manually labelled Ntr=1215 training tweets in the time span from January 1st, 2020 to May 1st, 2020. This training set was created using comments regarding different politicians. A uniform distribution of the three classes was given to prevent our learning model to become biased towards one class rather than the other. The tweets were divided in:

- 405 tweets of class *in favour* of the politician: comments expressing a feeling of trust and support towards the politician and his/her work or ideas.
- 405 tweets of class *in contrast* with the politician: comments expressing a feeling of disappointment with the decisions made by the politician, his/her ideas or behaviour. Often such comments were characterised by mockery, offensive terms and strong sarcasm.
- 405 tweets of class *neutral*: comments that report news headlines, neutral opinion tweets or comments related to third part characters. Not too polarised opinions or pieces of advice belonged to this category too.

The table 1.1 shows an example of manually labelled training tweets.

The next step we followed was the preprocessing of the labelled tweets: we removed all of duplicate tweets along with the characters that were not letters (punctuation marks, numbers and special characters). We also deleted elements such as link and emojis. Hashtags were not removed because they maintained valuable information regarding the tweets themselves. At the end of this phase, the tweet messages were made up of only alphabetic characters ready to be processed in the next phase.

The third phase focused on the text representation using the standard "Bag-of-Words" (BOW) approach. Its main aim was to transform the tweet messages from a set of strings to a set of vectors. This was done by following these stages:

1. *Tokenization*: consisted in transforming a string of alphabetic characters into a series of processing units called "tokens". According to the different types of N-grams chosen (where N typically varies between 1 and 2), the representation of the text tokens could change in this stage.
2. *Stop-words filtering*: consisted in removing words that were of little use for the aim of our experiment. These stop-words included articles, prepositions and conjunctions. We downloaded a list of Italian stop-words from the link `https://github.com/stopwords-iso/stopwords-it/find/master`.
3. *Stemming*: this stage is particularly useful for Indo-European languages because each token is reduced to its basic form. Words such as "bella" and "bello" ("beautiful" in Italian) were reduced to their root form "bell". In this case we used the stemmer provided by Snowball Tartarus for the Italian language.

Tweet text	Assigned label
@GiuseppeConteIT Presidente, purtroppo sto vivendo questa situazione in Spagna e le posso assicurare che avrei preferito stare in Italia che ha saputo gestirla meglio, sia per la trasparenza che per la Sua preoccupazione empatica. Avanti e Forza! *English translation: «President, unfortunately I am living this situation in Spain and I can assure you that I would have preferred to stay in Italy which has been able to manage it better, both for transparency and for your empathic concern. Come on and go!»*	**In favor**
@gualtierieurope Complimenti per il gran lavoro che state svolgendo per gli italiani. Avete il nostro sostegno. Lasciate perdere chi sa solo insultare e distruggere senza nessuna proposta intelligente. *English translation:«Congratulations on the great work you are doing for the Italians. You have our support. Forget those who only know how to insult and destroy without any intelligent proposal.»*	**In favor**
@robersperanza Il miglior vaccino sarebbe quello che elimina il rischio di avere ministri come lei. *English translation:«The best vaccine would be the one that eliminates the risk of having ministers like you.»*	**In contrast**
@AzzolinaLucia Brava Ministra! Il suo è ministero più difficile e bistrattato di tutti! I suoi precedessori hanno fatto solo macerie: non dimentico i 33 studenti per classe pretesi ai tempi della Gelmini! Oppure l'orrenda Buona Scuola della Giannini. In bocca al lupo! *English translation:«Good Minister! His ministry is the most difficult and mistreated of all! Its predecessors have only made rubble: I do not forget the 33 students per class required at the time of Gelmini! Or the horrendous Good School of Giannini. Good luck!»*	**In favor**
@matteosalvinimi Il movimento #BlackLifesMatters caro Salvini vale anche in Italia per gente come lei che non ha fatto altro che diffondere ideali di razzismo promuovendo la supremazia bianca e il cristianesimo in uno stato dichiaratamente democratico e laico. *English translation: «Dear Salvini, the #BlackLifesMatters movement is also valid in Italy for people like you who have done nothing but spread the ideals of racism by promoting white supremacy and Christianity in an openly democratic and secular state.»*	**In contrast**
@luigidimaio Cavolo ma sei davvero un grande... Noi invece tutti imbecilli. Menomale ci sei tu...con la tua laurea, la tua bella faccia... 😂😂😂😂😂 *English translation:«Damn but you are really great ... We are all idiots. Luckily there is you ... with your degree, your beautiful face ...»*	**In contrast (very sarcastic)**
@matteorenzi Raddoppiare il job acts, magari cambiando il nome in italiano, come sa sulla sua pelle, molti italiani non conoscono l inglese *English translation:«Double the job acts, perhaps changing the name to Italian, as you know on your skin, many Italians do not know English»*	**Neutral**
@sbonaccini Presidente, svolta green sulle auto elettriche. Subito incentivi regionali, stop ai combustibili fossili. Transizione energetica dev'essere una priorità! *English translation:«President, green turnaround on electric cars. Now regional incentives, stop to fossil fuels. Energy transition must be a priority!»*	**Neutral (advice)**

Figure 1.1: Examples of manually labelled tweets

4. *Stem filtering*: here we filtered the stems that were considered irrelevant in the training dataset for the supervised learning stage.

5. *Feature representation*: this step created a numeric vector in a supervised learning stage for each training tweet. Each element of these vectors was set to the product of the frequency and the weight of the corresponding stem if it was relevant. 0 if it wasn't.

In the next phase we compared the classification results obtained by different machine learning algorithms. All of these algorithms were executed using "Weka" (Waikato Environment for Knowledge Analysis). Using this application, we were able to test different methods for the text tokenization (Word tokenizer and N-gram tokenizer) and to use *TF-IDF* (Term frequency–Inverse document frequency) for the feature representation. More details regarding the results we achieved will be given in the following paragraphs.

1.2 Experimental comparisons

The experiments were performed using a 10-fold stratified cross validation (CV) procedure. Having a total of 1215 (N_{tr}) labelled tweets, at each iteration, the classification model was trained on about 1093 tweets, and tested on about 122 tweets. We repeated the 10-fold stratified CV twice, using two different seed values to randomly partition the data into folds. All the compared schemes included three phases: a first phase for transforming texts into numerical vectors of features (*relevant stems*), a second phase for the attribute selection and a third phase for the classification. We adopted BOW for the text representation and the classical machine learning classification models to classify the tweets. The experiments were made with the following classification algorithms: Support Vector Machine (SMO), Simple Logistic, Naïve Bayesian , Multinominal Naïve Bayesian , JRip , C4.5 decision tree , Random Forest and 1-Nearest-Neighbor. We evaluated these models in terms of accuracy, precision, recall, weighted F-measure and AUC. These metrics (except for accuracy and weighted F-measure) were evaluated for each of the three classes.

Chapter 2

Experiments

Initially we tested different classification models without an attribute selection phase. The text representation phase included all the classical BOW steps: tokenization using "1-gram" tokenizers, stop-words filtering, stemming and stems filtering. As said before, we adopted a 2x10-fold cross-validation to evaluate the achieved results.

Classifier	Class	F-Measure	Precision(%)	Recall(%)	AUC	Accuracy(%)	Weighted F-Measure
SMO	In Favor	0,64	63,9	64,1	0,7535	54,7325	0,5475
	In Contrast	0,51	55,05	47,5	0,671		
	Neutral	0,493	46,45	52,6	0,6095		
Simple Logistic	In Favor	0,6545	71,35	60,45	0,7895	55,1441	0,5525
	In Contrast	0,4745	55,3	41,6	0,6955		
	Neutral	0,528	45,3	63,3	0,6625		
Naive Bayes	In Favor	0,6265	74,35	54,1	0,793	53,4157	0,5395
	In Contrast	0,531	48,45	58,8	0,7085		
	Neutral	0,4605	44,75	47,45	0,6655		
Naive Bayes Multin.	In Favor	0,6235	58,75	66,4	0,7725	52,8807	0,5215
	In Contrast	0,53	49,8	56,55	0,6925		
	Neutral	0,4115	48,65	35,65	0,6525		
Jrip	In Favor	0,5655	47,15	71,7	0,704	45,1438	0,4225
	In Contrast	0,365	50,15	34,2	0,63		
	Neutral	0,3365	41,5	29,5	0,6		
J48	In Favor	0,558	71,35	45,8	0,692	48,7654	0,482
	In Contrast	0,3755	47,9	30,9	0,625		
	Neutral	0,5135	40,6	69,65	0,5985		
Random Forest	In Favor	0,646	64,65	64,55	0,8115	54,6914	0,5435
	In Contrast	0,461	57,35	38,5	0,713		
	Neutral	0,524	45,85	60,95	0,6645		
1-NN	In Favor	0,6095	57,9	64,45	0,745	50,6173	0,487
	In Contrast	0,333	54,8	24	0,6075		
	Neutral	0,518	43,8	63,45	0,6165		

Table 2.1: Results achieved by different classifiers with no attribute selection.

The best results were achieved by SMO and Simple Logistic, which presented the highest values for both accuracy and weighted F-measure. As a second step we included an attribute selection phase: the quality of each stem was evaluated by means of the *Information Gain* value and which was then used to rank the stems in descending order. We experimented different configurations for the number of features to select in the ranking.

Classifier	**None**	**Threshold$_0$**	**200 features**	**300 features**	**500 features**	**2000 features**
SMO	54,7325	55,9259	54,98	55,44	54,61	54,033
SimpleLogistic	55,1441	56,1317	55,85	56,01	55,48	55,0617
NaiveBayes	53,4157	55,3086	55,06	54,82	53,99	53,251
NaiveBayesM	52,8807	47,6954	50,29	50,99	50,79	51,893
Jrip	45,1438	47,05	47,3	45,61	45,19	43,8683
J48	48,7654	46,54	47,37	47,98	47,82	48,7243
RandomForest	54,6914	55,6	55,65	56,42	54,49	52,3457
1-NN	50,6173	55,31	54,28	54,49	51,44	44,6502

Table 2.2: Accuracies using different attribute selection configurations

Table 2.2 shows that the attribute selection phase increased the accuracy levels of most classifiers and the best choice was to adopt a ranker with a threshold value of 0. Indeed, the feature space was reduced from 3306 to only 97 attributes and the accuracy levels for SMO (55,9259%) and Simple Logistic (56,1317%) increased. The "Correlation-based Feature Subset Selection" ("CfsSubsetEval" in Weka) was also tested in the attribute selection phase. However the achieved results were not of much interest and have therefore been excluded from this report. In order to verify if any statistical differences exist among the results achieved by the 8 classification models, we also performed a statistical analysis. We applied the parametric statistical test *t-test* to each classifier: we generated a distribution consisting of the 20 values of the accuracies obtained by repeating the 10-fold cross validation twice. We selected Simple Logistic as the control algorithm and compared the values achieved by this model with the ones achieved by the other classifiers. Simple Logistic was declared to be statistically comparable to SMO, Naive Bayes, Random Forest and 1-NN. In table 2.3 the metrics of these classifiers are shown.

By analysing tables 2.2 and 2.3, we noticed that both SMO and Simple Logistic were the classifiers that achieved the best results, especially in terms of accuracy and weighted F-measure. In the following chapter we made further experiments on both these classifiers because the information we gathered so far did not allow us to reject either one.

Classifier	Class	F-Measure	Precision(%)	Recall(%)	AUC	Accuracy(%)	Weighted F-Measure
SMO	In Favor	0,6595	76	58,3	0,7585	55,9259	0,5515
	In Contrast	0,434	68,7	31,7	0,692		
	Neutral	0,5615	43,9	77,8	0,6405		
Simple Logistic	In Favor	0,66	76,55	58	0,793	56,1317	0,555
	In Contrast	0,439	65,8	32,95	0,6955		
	Neutral	0,565	44,5	77,4	0,65		
Naive Bayes	In Favor	0,636	73,95	55,8	0,7835	55,3086	0,546
	In Contrast	0,4415	66,9	32,95	0,6999		
	Neutral	0,5605	44,05	77,15	0,653		
Random Forest	In Favor	0,657	74,85	58,55	0,7935	55,5968	0,549
	In Contrast	0,432	65,6	32,25	0,7045		
	Neutral	0,5575	44,05	76,05	0,644		
1-NN	In Favor	0,649	74,65	57,4	0,7865	55,3087	0,545
	In Contrast	0,4295	66,4	31,75	0,6945		
	Neutral	0,5575	43,8	76,8	0,6345		

Table 2.3: Average results of the best classifiers with a InfoGain and a ranker threshold value of 0.

Figure 2.1: Stem-cloud of the attributes selected in the 2x10-fold cross-validation

Chapter 3

Online monitoring

The online monitoring phase took into consideration the period of time between May 1st 2020 and December 1st of the year 2020. During this time interval we searched for sudden increases in the total number of tweets written to all politicians. We decided to not consider peaks of tweets regarding *single* politicians, because the total number of these peaks would have become too high. This would have made it impossible to consider a single peak useful at all for the retraining of our classification model and would not have allowed us to clearly identify if there was any *concept drift*.

To detect a peak of tweets, we decided to use the following bilateral moving average that indicates whether of not the *i-th* day contains a peak:

$$Peak(i) = \begin{cases} True, & \text{if } Tweets(i) > 1.3 * \frac{\sum_{h=i-7}^{i+7} Tweets(h)}{15} \text{ and } Tweets(i) > 10000 \\ False, & otherwise \end{cases} \tag{3.1}$$

In the previous formula $Tweets(i)$ indicates the total number of tweets written to all the politicians on the *i-th* day of monitoring. The parameters were chosen because they provided the best trade-off between the number of peaks detected and the actual significance each of these peaks might have had for our model.

Once all the general peaks were identified, we then proceeded to group all those which were related to adjacent days. After this we searched for possible events that might have caused these increases in tweets. Since many politicians were taken into consideration, each of these events might have been related to different politicians and each peak itself might have been caused by multiple events. We finally decided to use a formula similar to the previous one to identify which politicians actually had a "personal" peak in the days in which a general one occurred. This allowed us to understand whether the real life events that were related to certain politicians actually brought an increase in the tweets they received. The formula we used to identify if politician j had a peak on the *i-th* day was the following:

$$Peak(i,j) = \begin{cases} True, & \text{if } Tweets(i,j) > 1.3 * \frac{\sum_{h=i-7}^{i+7} Tweets(h,j)}{15} \\ False, & otherwise \end{cases} \tag{3.2}$$

In this case, $Tweets(i,j)$ refers to the number of tweets politician j received on the *i-th* day of monitoring. The results of this entire process have been summarised in table 3.1.

General Peak Number	Starting date of peak	Duration (Days)	Politicians with personal peaks	Events
1	2020-05-20	1	Azzolina, Di Maio, Renzi, Raggi, Zingaretti	The Senate rejects the no-confidence motions on Bonafede [1]
2	2020-06-02	2	Berlusconi, Conte, Crimi, Meloni, Salvini, Tajani, Toti, Zingaretti	Italian Republic Day Speech given by Conte who asked the population to be united [2]
3	2020-06-13	1	Conte, Crimi, Di Maio, Gualtieri, Meloni, Speranza	First day of the Italian "General States" [3] Ongoing agreement for the vaccination campaign against COVID-19 [4] Conte replies to Miley Cyrus' tweet [5]
4	2020-07-21	1	Berlusconi, Crimi, Conte, Di Maio, Meloni, Renzi, Salvini	Conte signs an agreement for the Recovery Fund [6]
5	2020-07-30	1	Bonaccini, Meloni, Renzi, Salvini, Zaia	Senate agrees to trial of Salvini for the Open Arms case [7] Bonaccini gives a speech on the "MES". [8]
6	2020-08-11	1	Azzolina, Di Maio, Meloni, Renzi	Renzi says the vaccine should be compulsory for everyone [9] Di Maio authorizes INPS to publish his personal data [10] Azzolina says it wouldn't be a scandal to have lessons in B&Bs [11] Azzolina makes a grammatical error in a tweet [12] Raggi is re-nominated as Mayor of Rome [13] Meloni promises to build a tunnel from Calabria to Sicily [14]

General Peak Number	Starting date of peak	Duration (Days)	Politicians with personal peaks	Events
7	2020-08-17	1	Conte, Meloni, Salvini, Speranza, Tajani, Zaia, Zingaretti	All discos in Italy are closed [15]
8	2020-09-14	2	Azzolina, Berlusconi, Speranza, Toti, Zingaretti	Reopening of public schools in Italy [16]
9	2020-10-25	1	Berlusconi, Bonaccini, Conte, Di Maio, Gualtieri, Meloni, Raggi, Tajani	New Decree of the President of the Council of Ministers [17]
10	2020-11-01	1	Berlusconi, Toti	Toti tweets that retired people are not necessary for the national workforce [18]
11	2020-11-04	1	Cirio, Conte, Crimi, De Luca, Di Maio, Salvini, Zaia, Zingaretti	A new Decree of the President of the Council of Ministers is postponed [19]
12	2020-11-07	1	Berlusconi, Conte, Di Maio, Gualtieri, Renzi, Salvini, Zingaretti	Election of Joe Biden as US President [20] Resignation of the Commissioner for Health in Calabria [21]

Table 3.1: Description of the peaks of tweets that were identified.

From table 3.1 several interesting conclusions could be drawn:

- Not all peaks that were linked to single politicians were caused by a specific political event. These peaks may have also been caused by an increase in the number of posts a certain politician made or by posts that could be easily criticised.
- Not all Decrees of the President of the Council of Ministers (*DPCM*) were directly linked to increases in the number of tweets. For example the Decrees of the 7th, 13th and 18th October were apparently of no influence at all. In the next chapters we analysed the effects of these particular events on specific politicians.

In the following graph, the time series regarding the daily number of tweets the politicians received have all been plotted one on top of the other. This allowed us to further understand which were the politicians who received the largest share of tweets and to confirm the conclusions previously made.

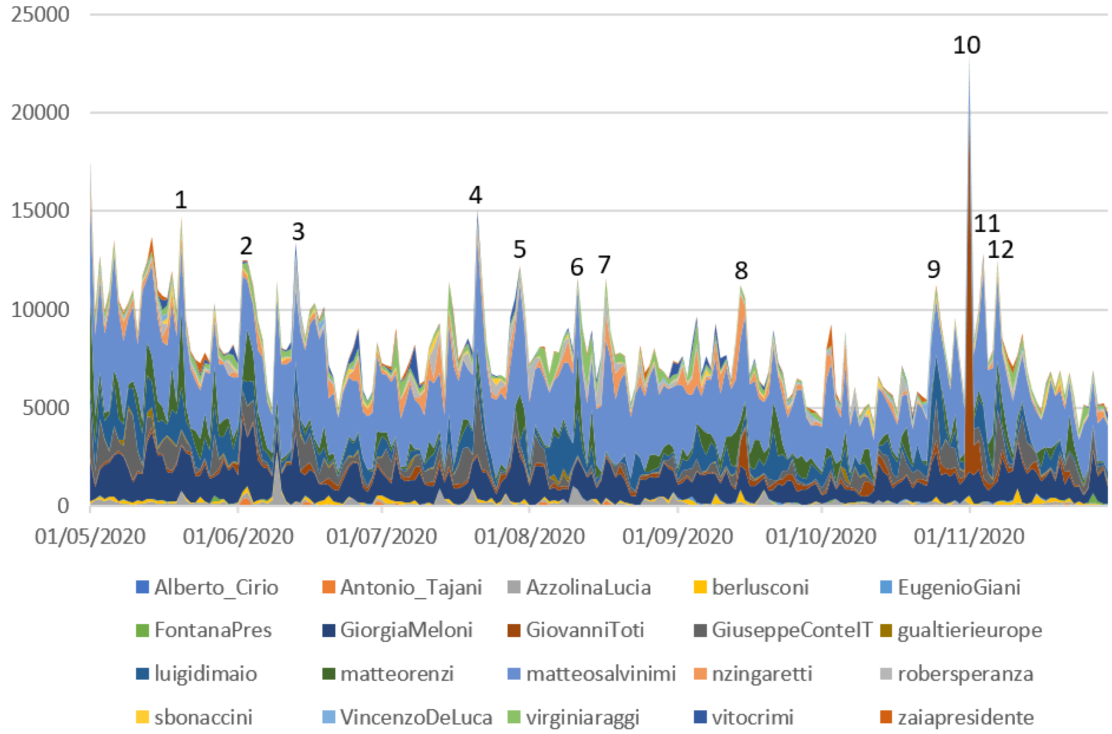

Figure 3.1: Number of daily tweets politicians received. The peaks have been identified using their corresponding number from table 3.1. The legend shows the Twitter accounts of all the politicians.

What we learnt from the graph in figure 3.1 were the following points:

- Formula (4.1) was capable of detecting all major peaks throughout the observed interval.
- The number of daily tweets the politicians received varied in a significant way. For example, Matteo Salvini received around half the number of total tweets while Alberto Cirio and several others only received tens of daily tweets on average.
- The average number of total tweets also varied in an important manner. In May this number was around 10000 while in October and at the end of November it decreased to just over 5000.
- By observing peaks 2 and 9, we noticed that there were some events that allowed many politicians to obtain personal peaks.

- ”General” peaks could also be strongly influenced by ”personal” peaks some of the politicians obtained. The most relevant example of this type of peak is number 10. In this case the tweet Giovanni Toti made received an enormous amount of criticism for a post he made. However, this criticism did not last long as there was an immediate decrease in the number of replies he received. This behaviour is typical of social networks such as Twitter and underlines the power they may have to influence peoples' lives and opinions, especially on subjective issues such as politics.

3.1 Concept Drift

After analysing all the peaks that occurred during the monitoring interval, we wanted to see if in the tweets we had scraped there was any *concept drift*. This term is used in Machine Learning to indicate possible unpredictable changes that might appear over time in the data we are trying to study. If there were any evident signs of *concept drift*, the optimal model used, in our case to classify the tweets, might have become less effective after a certain period of time. To test this, 200-300 new tweets were therefore taken to be used as new test sets from the politicians who obtained a personal increase for each of the 12 peaks we identified. Each of these new datasets had around 10% positive tweets, 65% negative tweets and 25% neutral tweets. These test sets were kept unbalanced in order to actually measure the ability the models had to classify real unbalanced test sets. We then sampled 20 tweets for each class (*in favour*, *in contrast* and *neutral*) from each peak to be added to the initial training set with the labelled tweets from January to May. This way we allowed the training sets to be updated but also prevented the classifiers from becoming biased towards one class rather than the other.
To study the effects of *concept drift*, we decided to compare the results obtained by 3 different learning models while classifying the tweets from the 12 identified peaks:

- The*Static* model, which did not perform any update to the initial training set.
- The *Incremental* model, which inserted new labelled tweets from a certain event into the training set after the event had been analysed.
- The *Sliding Window* model, which also allowed the training set to be updated like the *incremental* model but it removed the oldest tweets from this data set in order to maintain the number of total tweets constant over time.

If there was any *concept drift* in the analysed tweets, the *static* model would have decreased the quality of its results the most because it could not be aware of the new topics that would appear over time.

The results obtained by classifiers SMO and Simple Logistic in this case were reported in the following graphs.

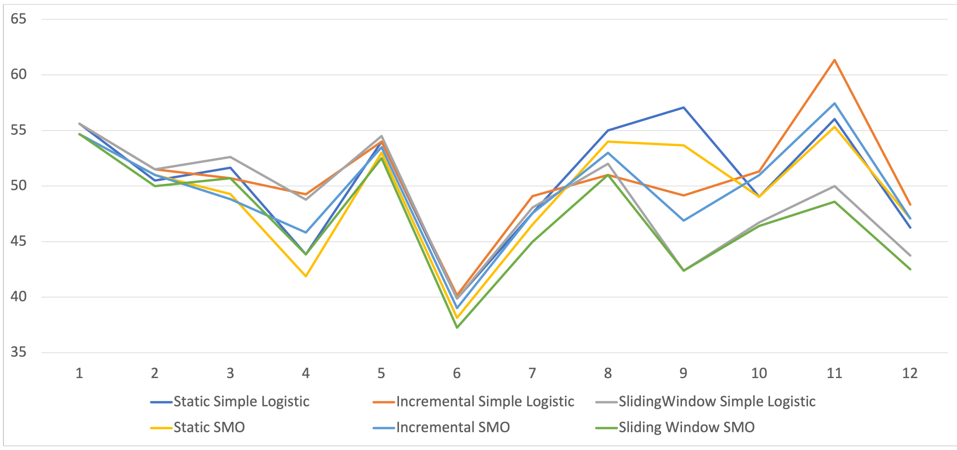

Figure 3.2: Accuracy for both SMO and Simple Logistic with all 3 considered models

Figure 3.3: Weighted F-measure for both SMO and Simple Logistic with all 3 considered models

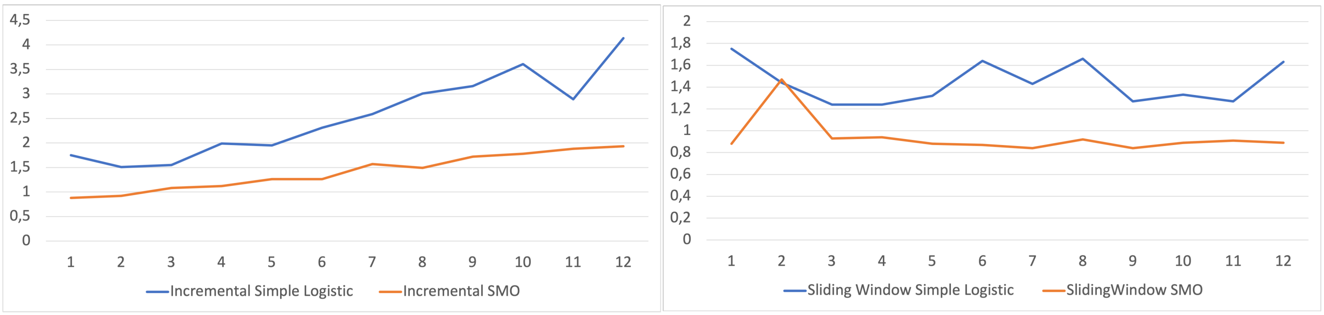

Figure 3.4: Time taken to build models for both SMO and Simple Logistic. On the left using the *Incremental* model and on the right using the *Sliding Window* model.

While observing the series in figures 3.2 and 3.3, we noticed that:

- The *Sliding Window* model was generally less accurate than the other two models for both SMO and Simple Logistic.
- By comparing the *Static* models of both classifiers in figures 3.2 and 3.3, we noticed that Simple Logistic generally obtained better results. The same thing could also be said for the *Incremental* models.
- While comparing all three models applied to SMO amongst themselves, we realised that the *Incremental* model was not always better than the other two, even if it could count on a much larger training set. In some cases it even achieved results that were much worse than the *Static* model. We also observed that this was also true for *Simple Logistic*.

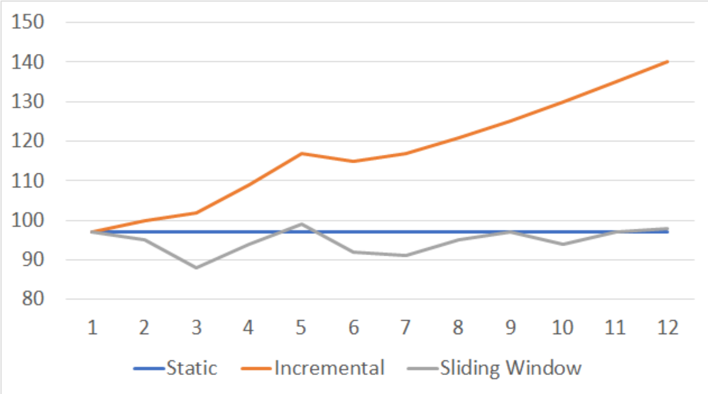

Figure 3.5: Number of attributes selected for each model using InfoGain and Ranker with threshold equal to 0

Moving on to figure 3.4, there was no room for any doubts: SMO was clearly the fastest classifier, reducing the time to build the model up to 50%. Looking at the graph on the left corresponding to the *Incremental* model, it was also very clear that SMO did not increase the time to build the model with the same speed of Simple Logistic. This surely provided an important clue in favour of it.

Figures 3.2 and 3.3 also allowed us to conclude that there was no evident sign of any *concept drift*. The Static model appeared not to have suffered in any way the passing of time. Only in a few cases the Incremental model turned out to be better. In figure 3.5 we also noticed that the *Incremental* model very quickly increased the amount of attributes it took into consideration. This did not result in an increase in the quality of the results obtained but it *did* increase the time it took to obtain them. The graph regarding the time it took for the *Static* model to build the classifying model has been excluded because it remains constant for both SMO (0,9 seconds) and Simple Logistic (1,75 seconds). These values were in any case much lower than the values of the *Incremental* model. This was a crucial point for our analysis: it allowed us to conclude that not only there was no evident proof that the *Static* model obtained worse results than the *Incremental* but also it was much faster.

3.2 Stop-Words Filtering

The analysis we made in the previous section allowed us to obtain some very interesting information. However we were not completely satisfied with the achieved results. We tried to make several modifications to our overall Data Mining process to increase the accuracy and weighted F-measure of our classifiers.

We first tried to remove the stemming of the tokens and to replace the Word-tokenizer by using a 2-gram and 3-gram tokenizer. We executed a 2x10 Cross-Fold validation for all these options but each time the results were worse than the original method.

We finally decided to not remove the stop-words from the tweets we analysed. The following table contains the results obtained after executing a 2x10 cross-fold validation without any attribute selection.

We immediately noticed a significant improvement compared to table 2.1. This lead us to further investigations: once again we tried to search for the best attribute selection algorithm. In the following table once again we reported the accuracies obtained by the different algorithms while operating with different classifiers.

A significant improvement was brought by not removing the stop-words if we compare the data to table 2.2. In this case not using any attribute selection included 3452 attributes while InfoGain with a ranker threshold=0 included 119 attributes, instead of only 97. Once again this last attribute selection algorithm proved to be the best option. SMO and Simple Logistic were able to obtain the best accuracies and weighted

Classifier	Class	F-Measure	Precision(%)	Recall(%)	AUC	Accuracy(%)	Weighted F-Measure
SMO	In Favor	0,641	63,3	64,9	0,7635	55,144	0,552
	In Contrast	0,5185	54,15	49,75	0,681		
	Neutral	0,494	48,1	50,75	0,632		
Simple Logistic	In Favor	0,655	74,1	58,65	0,791	57,4898	0,577
	In Contrast	0,5155	56,2	47,65	0,713		
	Neutral	0,561	48,65	66,15	57,4898		
Naive Bayes	In Favor	0,61	58	64,35	0,7625	53,4568	0,526
	In Contrast	0,5445	48,8	61,5	0,7205		
	Neutral	0,2435	54,65	34,55	0,712		
Naive Bayes Multin.	In Favor	0,637	61,2	66,3	0,785	55,22635	0,547
	In Contrast	0,5495	50,85	59,8	0,711		
	Neutral	0,4555	53,4	39,65	0,68		
Jrip	In Favor	0,5325	45,55	64,85	0,6665	42,8807	0,406
	In Contrast	0,4205	40,65	43,8	0,5985		
	Neutral	0,266	41,35	20	0,5895		
J48	In Favor	0,531	58,05	48,9	0,679	48,60085	0,487
	In Contrast	0,445	48,3	41,2	0,62		
	Neutral	0,483	42,65	55,65	0,6015		
Random Forest	In Favor	0,6515	65	65,3	0,808	56,3786	0,564
	In Contrast	0,549	53,8	56,05	0,731		
	Neutral	0,4895	50,15	47,8	0,68		
1-NN	In Favor	0,5935	49,25	74,7	0,745	47,9424	0,46
	In Contrast	0,452	48,55	42,2	0,642		
	Neutral	0,3335	43,9	26,95	0,576		

Table 3.2: Results achieved by different classifiers with no attribute selection and stop-words filtering.

Classifier	None	Threshold_0	200 features	300 features	500 features	2000 features
SMO	55,144	59,63	59,18	58,06	57,08	54,98
Simple Logistic	57,4898	59,14	58,93	58,61	58,4	58,19
NaiveBayes	53,4568	58,6	58,03	57,25	56,09	53,53
Naive Bayes M.	55,22635	55,1	54,32	53,75	53,71	55,68
Jrip	42,8807	45,53	47,38	44,91	44,86	43,96
J48	48,60085	50,91	50,91	50,91	50,54	48,76
Random Forest	56,3786	55,03	56,1	56,34	55,68	54,69
1-NN	47,9424	49,68	49,96	48,68	47,53	45,14

Table 3.3: Accuracies over different attribute selection configurations without any stop-word filtering

F-measures. We then performed the same parametric statistical test as described in chapter 3, this time using SMO as the control algorithm. The results indicated that Simple Logistic and Naive Bayes were both statistically similar to SMO. Their results we obtained after having executed once again a 2x10 cross-fold validation, with InfoGain and ranker with a threshold=0, have been inserted in the following table.

At this stage we then repeated another set of statistical tests in which we compared the same classifiers with and without the stop-words filtering. In all these tests, the case in which there was no filtering proved to be statistically better.

Classifier	Class	F-Measure	Precision(%)	Recall(%)	AUC	Accuracy(%)	Weighted F-Measure
SMO	In Favor	0,657	76,75	57,4	0,7575	59,63	0,5955
	In Contrast	0,535	63,8	46,05	0,7255		
	Neutral	0,5965	49,3	75,45	0,689		
Simple Logistic	In Favor	0,657	76,35	57,65	0,7955	59,14	0,5935
	In Contrast	0,546	60,85	49,5	0,7455		
	Neutral	0,578	49,1	70,25	0,699		
Naive Bayes	In Favor	0,6345	73,85	55,55	0,773	58,6	0,5875
	In Contrast	0,5485	60,5	50,25	0,7465		
	Neutral	0,5795	49,4	70	0,7155		

Table 3.4: Results achieved by the classifiers which were statistically comparable to SMO, with no stop-words filtering and InfoGain+Ranker with threshold=0 as attribute selection algorithm.

In figures 3.6 and 3.7, we graphed the accuracy and weighted F-measure levels for SMO using all 3 learning models (Static, Incremental and Sliding Window), with and without stop-word filtering while classifying the tweets of all the 12 identified peaks.

Figure 3.6: Accuracy of SMO with all 3 learning models (Static, Incremental and Sliding Window) with and without stop-word filtering

The graphs in figures 3.6 and 3.7 allowed us to prove that SMO actually improved the quality of its results without the stop-word filtering. The accuracy of the classification of the tweets of the 12 peaks for example increase by an average of 7,6%. They also allowed us to conclude that once again the *Static* model is still very effective if compared to the other 2 models. All considerations made in the previous section are therefore still valid.

By looking at all the data in table 3.4, figures 3.6 and 3.7 and all the observations in the previous section regarding the time it took to build the models, we decided to choose SMO as our final classifier, the *Static* model as our learning model, *InfoGain+Ranker with a threshold=0* as our attribute selection algorithm and *Word-tokenizer* as our tokenizer.

Before moving on to the next section, we decided to make one last test. We wanted to see what would have happened if in our *Static* model we also included the balanced data set with the tweets from all the 12 peaks we previously identified. If there was a noticeable improvement in the results obtained, we could use this new "updated" model to classify future tweets.

Figure 3.7: Weighted F-measure of SMO with all 3 learning models (Static, Incremental and Sliding Window) with and without stop-words filtering

Classifier	**Class**	**F-Measure**	**Precision(%)**	**Recall(%)**	**AUC**	**Accuracy(%)**	**Weighted F-Measure**
SMO	In Favor	0,657	76,75	57,4	0,7575	59,63	0,5955
	In Contrast	0,535	63,8	46,05	0,7255		
	Neutral	0,5965	49,3	75,45	0,689		
SMO updated	In Favor	0,6865	75,4	63	0,7815	61,94	0,6206
	In Contrast	0,573	64,6	51,45	0,7415		
	Neutral	0,6025	52,15	71,35	0,699		

Table 3.5: Final comparison between the results achieved by SMO with only the initial set of tweets and with the tweets from the 12 identified peaks.

The "updated" version of SMO allowed us to actually obtain better results in terms of accuracy and weighted F-measure. This might also have been due to the fact that there was a significant increase in the size of the training set. This version had a training set of 2010 tweets (the initial *Static* model had only 1215), it took around 3,2 seconds to build the model and it considered 166 attributes. In any case, the results we achieved with this combination of algorithms allowed us to obtain far better results than any other.

Chapter 4

Analysis of the classified tweets

In this chapter we described the results obtained by using SMO to classify all the remaining tweets in our dataset. Due to the number of considered politicians and the variety of situations they may have experienced, we obtained many different graphs. In the following pages we included only those who were thought to be more significant. The data is presented with a summarised weekly format because it allowed the graphs to be read more easily.

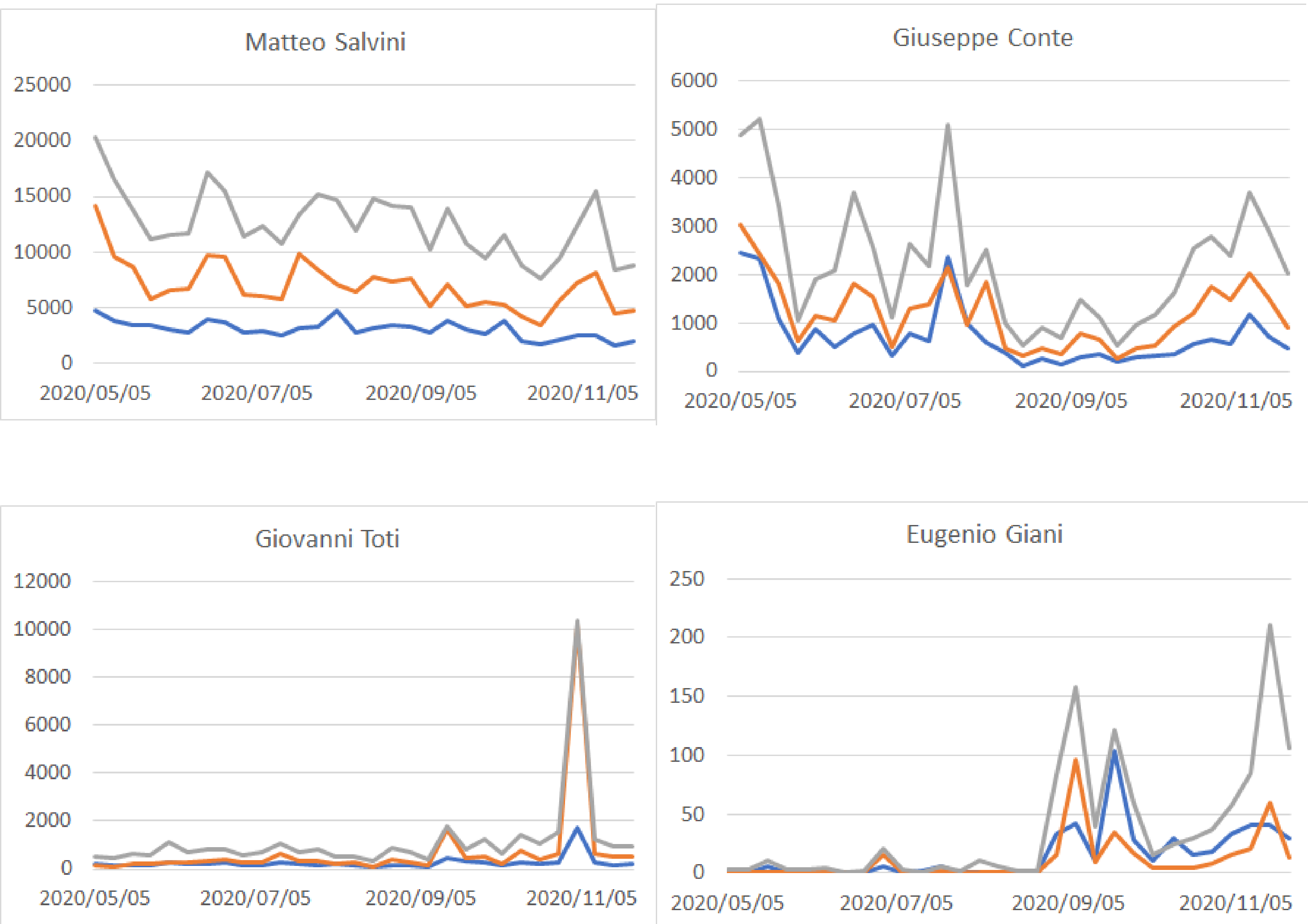

Figure 4.1: Number of weekly tweets different politicians received. In blue the number of tweets in favour, in orange the number of tweets in contrast and in grey the number of neutral tweets.

The graphs in figure 4.1 were very different from one another. The following observations were made:

- The politicians we analysed came in a large variety both in the number of followers and comments they received. Just by looking at the y-axis of each of these graphs we can notice this.

- Generally all politicians receive many more negative and neutral comments than positive ones. There were however some events or tweets the politicians themselves posted that could reverse this trend. An example of this was Eugenio Giani in September.

- Of course there were many cases in which the politicians received peaks of messages of hatred and contrast. The tweet Giovanni Toti posted in November 2020 is an important example of this and will be analysed again in the next pages.

- As previously stated, Matteo Salvini was the politician who received on average the highest number of tweets, even if there was a decreasing trend in these numbers. Another very interesting fact was that the tweets happened to maintain a very precise distribution amongst the different classes. This happened even when for example he was sentenced to trial for the Open Arms case on the 30th of July.

- Giuseppe Conte, who was certainly one of the politicians who received the most attention from the media, did not have any period of time in which the negative comments increased particularly. There was a peak of *neutral* comments at the beginning of May, at the same time when the first lockdown in Italy ended. Another peak of *in favour* and *neutral* tweets occurred around the 21st of July when he signed the agreement for the Recovery Fund. Many people evidently had high hopes for this Fund which was seen as a sort of "silver lining" that could help contrast the Coronavirus. Instead during September and October, Conte generally received very few replies. Nevertheless, when the second national lockdown approached, these replies increased very quickly once more.

All these considerations made it very clear that more experiments needed to be done.

4.1 User Stance Analysis

We proceeded to executing a different type of analysis that focused on the single users and their daily tweets to the politicians. This was done so that we could improve our general understanding of the situation. More precisely, our goal was to understand better the *public opinion* towards every politician in a certain period of time. We made sure that this value did not depend on the number of *tweets* but on the number of *social users* in favour or in contrast towards a certain politician. This way less importance was given to any social user, which could be a human or a bot, who wrote many tweets of the same class during the course of the same day.

We defined a value called *UserScore* for every day for every user for every politician. The way we calculated the *UserScore* for user i, politician j and day t is the following:

$$UserScore(i,j,t) = \frac{Tweets\_In\_Favour(i,j,t) - Tweets\_In\_Contrast(i,j,t)}{Tweets\_Total(i,j,t)} \tag{4.1}$$

where *Tweets_In_Favour(i, j, t)* is the number of tweets user i wrote in favour of politician j on day t. *Tweets_In_Contrast* and *Tweets_Total* refer respectively to the number of tweets in contrast and total tweets written. The *UserScore* was bound to be between the values of -1 and +1. A user who was clearly in favour of a politician on a certain day will obtain a *UserScore* equal to 1. On the other hand, a user who wrote many tweets in contrast to the politician would obtain a value of -1. This was the same *UserScore* a user would receive even if he/she had just written 1 contrasting tweet to a politician on that day.

At this point, we used the *UserScore* of every user to calculate their *UserStance* which was defined as follows:

$$UserStance(i,j,t) = \begin{cases} +1, & \text{if } UserScore(i,j,t) > \sigma \\ 0, & \text{if } -\sigma <= UserScore(i,j,t) <= \sigma \\ -1, & \text{if } UserScore(i,j,t) < -\sigma \end{cases} \tag{4.2}$$

We chose a value of $\sigma=$ 0,2 , following the proposal made in [22]. We finally determined the overall *Public Opinion* towards politician j on day t in the following manner:

$$PublicOpinion(j,t) = \sum_{i=1}^{N} UserStance(i,j,t) \tag{4.3}$$

, where N was the total number of users who replied to politician j on day t.

What truly allowed this formula to be more effective than the simple difference between *in favor* and *in contrast* tweets to a politician was that, in this case, very "talkative" or "noisy" users did not affect the overall polarity of the *Public Opinion*. This allowed us to understand what the true *Public Opinion* was towards a certain politician and how it changed over time.

In figure 4.2, we reported the *Public Opinion* towards the same politicians of figure 4.1.

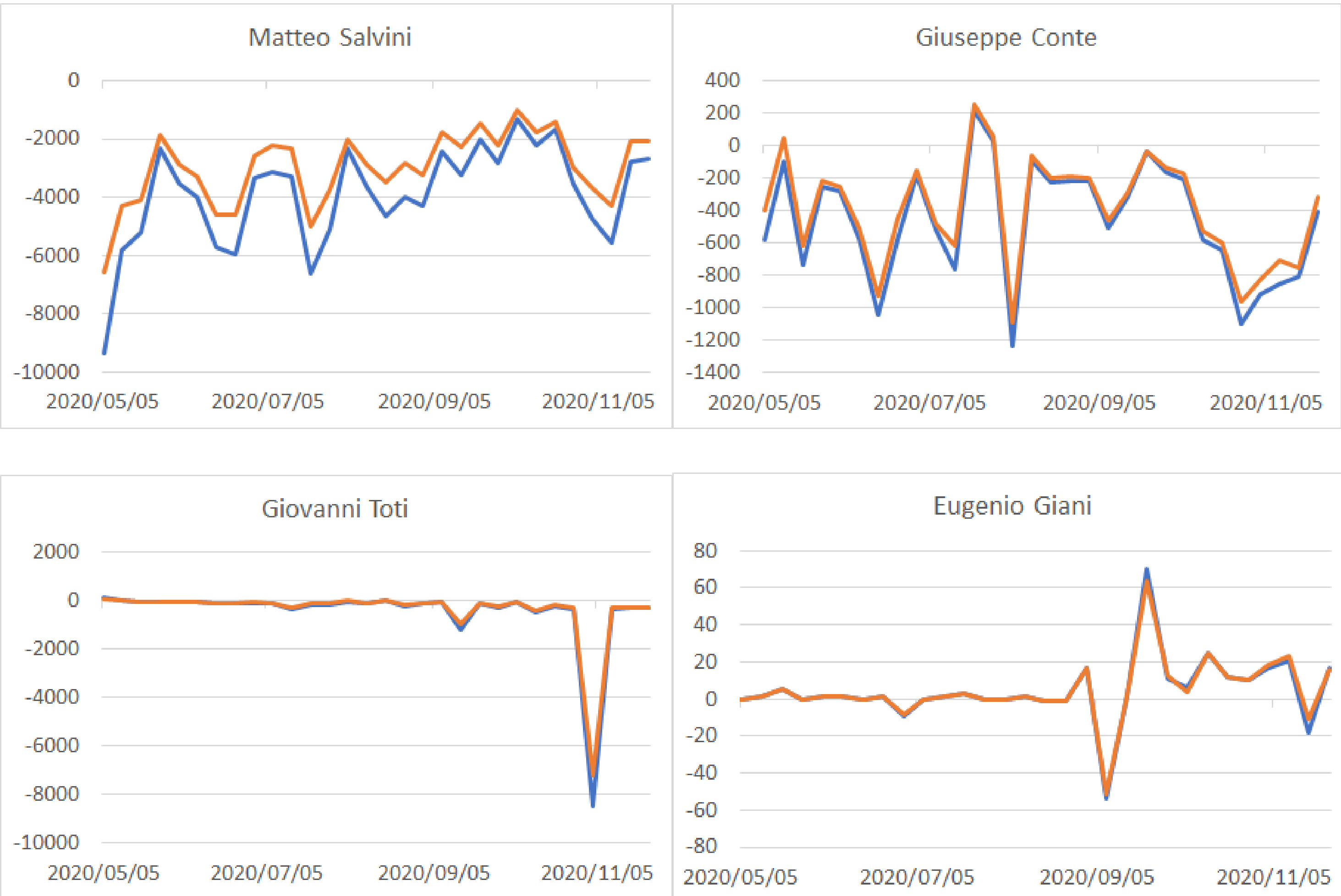

Figure 4.2: Weekly data of different politicians. In blue the difference between the number of tweets in favour and in contrast. In red the *Public Opinion* of each politician calculated using 4.3.

Once again, we noticed a very different behaviour of the graphs depending on the politician they referred to. The difference was both in terms of quantity and quality of the results. Several observations were made for figure 4.2 because it was the core of all our research. We first started analysing the blue series:

- Matteo Salvini was the politician who obtained the worst feedback. The difference between the tweets in his favour and in contrast with him was impressive. At the beginning of May it nearly reached -10000 units. The highest value this series reached was around -2000 in October 2020.
- Giuseppe Conte had a much smaller number of replies than Salvini. These tweets were however more equally distributed between the *in favour* and *in contrast*. The values of the blue time series were much closer to 0. The lowest values were obtained in mid-June, the beginning of August and in October. In June, Conte was busy with the Italian *General States* and the agreement for the vaccination campaign. What most people disliked however was the retweet he made of Miley Cyrus' vaccination campaign. In the second case, Conte at the beginning of August had just extended the state of emergency for the COVID-19 disease and received accusations from other important politicians, such as Giorgia Meloni and Salvini himself. Conte replied by saying "*I find it dangerous that a climate of mistrust that does not match reality is instrumentally fueled.*". Lastly, in October Conte had once again revisited the Italian policies to cope with the Coronavirus and had introduced a second lockdown phase. The positive peak Conte had on the 21st of July was mainly due to the fact that he had just signed the agreement for the Recovery Fund.
- Eugenio Giani had a very particular *Public Opinion* curve. From May to the beginning of September he received very few tweets, as we could also notice in figure 4.1. In September, before the regional elections in Tuscany, he made several tweets on agriculture, the healthcare system and new investments to make in Tuscany. Many people criticised these statements and this brought to the first rapid fall. After Giani won the regional elections, he posted several tweets in which many people congratulated him. This caused the positive peak in September. From the elections onwards, the *Public Opinion* towards Giani continued to maintain a positive trend. The interesting pattern we described in this case had a similar shape of the heart activity during an electrocardiography. It highlighted how quickly opinions of people who are in favour of a politician can be replaced by those of people in contrast. Other politicians, such as Conte in July and August, also experienced this pattern of alternating peaks and valleys. On the other hand, Toti's valley in November was the perfect example in which the general opinion on a certain subject was only unilateral. Fortunately for him, this did not influence the comments people made in the following weeks.

Having made all these observations, we then tried to answer the following question: when the *Public Opinion* and the blue time series were far off from one another, was it because the users repeated comments with the same (negative) label or was it because they just made more tweets that could be of a different labels? In the first case, when a peak of contrasting tweets occurred, many users would want to repeat their comments in contrast to the politician multiple times. This way, the difference between the number of positive comments and negative ones (blue series) would increase negatively, while the *Public Opinion* would not do so because the lowest value each *UserStance* could obtain was -1 (not -2, -3, ...). In the second case, the users would empirically be more likely to end up having a *UserStance* equal to 0 and not to +1 or -1, thus resulting in a general decrease in module of the *Public Opinion* series.
What we *did* notice however was another interesting fact: generally, if user i had made at least 1 comment in contrast to politician j on day t, the corresponding *UserStance* value would most probably be equal to -1. In fact, the only way for this user to obtain a *UserStance* value different to -1 was if he/she wrote 2 or more *neutral* comments or comments *in favour* of the politician during the same day. This was empirically difficult to obtain, unless the users had the opportunity to have many different interactions with the same politician on a single day. This behaviour, along with the fact that normally users did not repeat their opinions multiple times, explained why in normal circumstances the 2 time series in figure 4.2 were very close to one another.
We tried to make several tests on the tweets Matteo Salvini received at the beginning of May to answer the question we previously asked ourselves. To make a comparison, we also made the same tests on the tweets Giovanni Toti received in November because also in this case, the politician had an important negative peak.

The results we obtained by analysing Toti's negative peak were the following: of all the people (≈ 1380) who wrote at least 1 *in favour* comment, 81,3% (≈ 1130) obtained a *UserStance* value equal to +1. 3,0% (≈ 40) of these users obtained a *UserStance* value of -1. Of all the people (≈ 7840) who wrote at least 1 *in contrast* comment, 97,6% (≈ 7650) obtained a *UserStance* value of -1 while only 0,05% (4) obtained a value of +1. What this means is that there was no significant overlapping between the two types of users: basically none of the users who wrote contrasting tweets to Toti obtained an *in favour UserStance*. The same is more or less also valid for those who wrote tweets *in favour* of Toti.

We now turned our attention to the graph regarding Matteo Salvini. In this case the results were much less clear: of all the people (≈ 1570) who wrote at least 1 *in favour* comment, only 50,3% (≈ 790) obtained a *UserStance* value equal to +1. 13,6% (≈ 210) of these users obtained a *UserStance* value of -1. Of all the people (≈ 3760) who wrote at least 1 *in contrast* comment, 86,8% (≈ 3270) obtained a *UserStance* value of -1 and 1,1% (≈ 40) obtained a value of +1.
In Toti's case, the percentage of people who wrote at least 1 *in contrast* comment but then did not obtained a *UserStance* equal to -1 was around 2,4%. This indicates that very few people who wrote at least 1 *in contrast* comment also tweeted *neutral* or *in favour* ones. Instead the people who wrote at least 1 *in favour* but did not obtain a *UserStance* equal to +1 were 18,7%. These numbers in Salvini's case reached 13,2% and 49,7% respectively. This clear increase allowed us to state that a much larger percentage of Salvini's followers wrote tweets that were classified in different labels. This was the reason to why the difference between the *UserStance* series and the blue series was so wide. The social users did not repeat the same messages over and over again but they interacted multiple times with Salvini, expressing different types of opinions. This is most probably because the people who replied to Salvini were not only more numerous in general, but they were probably more used to interact with his many posts. Salvini's habit of tweeting multiple times throughout the same day probably gave his followers a chance to write many messages that could be of different nature and that could be classified not only into a single class. Toti's peak on the other hand was caused by only one post that received a huge quantity of criticism. The users did not change their opinion on that single post. This means that the percentage of the many users who wrote *in contrast* tweets and obtained a *UserStance* equal to -1 was very high (97,6% as written before). This caused the *Public Opinion* to become very negative. The difference with the blue time series can therefore be attributed mainly to users who wrote many negative tweets, which is different to Salvini's case.

Chapter 5

Conclusions

The analysis made in this report was based on the replies several Italian politicians received during the period of time between the 1st January and the 1st December 2020. We used part of the data as a training set and implemented the classical machine learning process for identifying the best way to classify the remaining tweets into 3 different classes (*In favour*, *In Contrast* and *Neutral*). We then identified several peaks in the number of tweets the politicians received and used some of those tweets for the *Online Monitoring* phase. This phase had the goal to study if there was any *concept drift* throughout the course of the year and to eventually select the best learning model to deal with this problem. Further research on the *stance* of single users allowed us to conclude that the *Public Opinion* on certain politicians and topics has a very intricate structure. Politicians, because of their social position, can receive many different replies which can come in a wide variety, both in terms of content and quantity. Periods of heavy criticism can come immediately before or after periods of neutrality and sometimes even favouritism. We also were able to detect cases in which the social users duplicated their opinions many times throughout the course of a single day. The *User Stance* analysis allowed us for example to identify the cases in which the number of tweets in contrast to a politician became very high not because the politician had many followers but because new users would interact for a short period of time with him or her. Giovanni Toti experienced such an increase because of a controversial tweet he posted in November 2020. This example and many others were explained in detail and allowed us to determine several possible responses the online public may have to political events and tweets published by the politicians themselves.

The peaks of *in contrast* tweets that Matteo Salvini and Giovanni Toti received were 2 very interesting examples that allowed us to really grasp the complexity of the world of social networks. They also revealed the importance of the *UserStance* analysis we made. It gave us the chance to understand better the peaks of comments the politicians had received and, more importantly, the nature of the messages of these tweets.

The research of the *Public Opinion* on politicians generated many interesting results and in the future may become very useful to understand the perception of many other politicians. It may be helpful to also lengthen the observed period of time to compare the perception of the same politicians after the Coronavirus pandemic.

References

1. https://www.repubblica.it/politica/2020/05/20/news/giorno_sfiducia_bonafede-257126918/

2. https://newsmondo.it/messaggio-conte-2-giugno-2020/politica/

3. https://www.ilmessaggero.it/economia/news/stati_generali_diretta_roma_conte_ultime_notizie_13_giugno_2020-5285543.html

4. http://www.salute.gov.it/portale/nuovocoronavirus/dettaglioComunicatiNuovoCoronavirus.jsp?id=5572

5. https://www.rainews.it/dl/rainews/articoli/Miley-Cyrus-scrive-a-Conte-aderisci-a-Global-Goal-United-for-the-Future-contro-coronavirus-8fd30c9b-90e5-4d79-ab3b-29a2674e6dd1.html

6. https://www.corriere.it/politica/20_luglio_21/recovery-fund-conte-piano-ambizioso-cosi-non-servira-mes-46d00960-cb09-11ea-bf7a-0cc3d0ad4e25.shtml

7. https://www.corriere.it/politica/20_luglio_30/salvini-va-processo-il-caso-open-arms-senato-concede-l-autorizzazione-procedere-37b84b96-d27e-11ea-9ae0-73704986785b.shtml

8. http://www.regioni.it/ue-esteri/2020/07/30/mes-bonaccini-serve-per-investire-su-sanita-di-qualita-616923/

9. https://www.corriere.it/politica/20_agosto_11/intervista-renzi-conte-vaccino-covid-obbligatorio-crisi-governo-5d09055e-db91-11ea-abc9-41b5baff53c0.shtml

10. https://www.askanews.it/politica/2020/08/11/bonus-deputati-di-maio-autorizzo-inps-a-diffondere-miei-dati-pn_20200811_00031/

11. https://www.secoloditalia.it/2020/08/azzolina-ne-spara-una-dietro-laltra-lezioni-nei-bb-salvini-la-fa-nera-e-lei-corre-ai-ripari/

12. https://www.ilgiornale.it/news/cronache/l-infrazione-dei-ladri-gaffe-azzolina-1882729.html

13. https://www.lastampa.it/roma/2020/08/11/news/roma-raggi-si-ricandida-e-spiazza-il-pd-il-m5s-archivia-il-limite-dei-due-mandati-1.39182477

14. https://www.ilmessaggero.it/italia/ponte_stretto_di_messina_tunnel_sotterraneo_progetto_costo_conte_news-5398230.html

15. https://www.fanpage.it/attualita/discoteche-chiuse-in-tutta-italia-da-lunedi-17-agosto-la-decisione-del-governo/

16. https://tg24.sky.it/cronaca/approfondimenti/scuola-riapertura-settembre

17. https://www.corriere.it/cronache/20_ottobre_25/nuovo-dpcm-covid-25-ottobre-2020-testo-definitivo-a71e3102-16a4-11eb-b530-8ca6e758b252.shtml

18. https://www.lastampa.it/cronaca/2020/11/01/news/i-morti-di-covid-sono-soprattutto-pensionati-non-indispensabili-l-autogol-su-twitter-di-toti-1.39487955

19. https://www.repubblica.it/politica/2020/11/04/news/scheda_dpcm_covid-273036534/

20. https://www.ilpost.it/2020/11/07/vittoria-biden-sconfitta-trump/

21. https://www.ilpost.it/2020/11/07/cotticelli-calabria-piano-covid/

22. A. Bechini, P. Ducange, F. Marcelloni and A. Renda, ”Stance Analysis of Twitter Users: the Case of the Vaccination Topic in Italy” in IEEE Intelligent Systems, vol. , no. 01, pp. 1-1, 5555.