UNIVERSITÀ DI PISA

Artificial Intelligence and Data Engineering

Data Mining and Machine Learning Project

GREENPASS Sentiment Analysis

Cancello Tortora Giuseppe
Macrì Armando

Academic Year: 2021-22

Summary

1 Introduction

The Green pass - EU digital COVID - certificate was born on the proposal of the European Commission to facilitate the free and safe movement of citizens in the European Union during the COVID-19 pandemic. It is a digital and printable (paper) certification, which contains a two-dimensional barcode (QR Code) and a qualified electronic seal. In Italy, it is issued only through the national platform of the Ministry of Health. The Certification attests to one of the following conditions: having had the COVID-19 vaccination, being negative on the molecular or rapid antigen test in the last 48 hours, being cured of COVID-19 in the last six months. The introduction of this document has unleashed various reactions in the Italian population (even to European) even triggering many strikes in the entire Italy.

The pandemic has been a topic of conversation that involved all strata of society. Dominating headlines in all media forms while social media for have not been different. Twitter for instance has had conversations related to COVID-19 dominating the trends, with opinions supporting or opposing measures taken by different governments to tackle the pandemic. A popular tactic amongst governments across the globe has been to lockdown their economies, to foster social and physical distancing in a bid to curb the spread of the virus.

Once the vaccination campaign has begun, the European countries decided to adopt a particular certificate with the name of EU Digital Covid Certificate, known in Italy as 'green pass' or 'certificazione verde' to allow a progressive reopening of all activities and economic recovery.

Without work commitments, or the license to work from home, people have taken to the internet to express a myriad of opinions, frustrations and emotions concerning the situation.

1.1 Goal of the project

Understanding the sentiment and tone conveyed in a text is really vital especially. The goal of this project is to carry out an analysis on the opinion about the Green pass, debate very popular nowadays.

The project consists in retrieving what is the popular opinion about Green pass, i.e., if people are favorable or contrary to the adoption of this measure by the Government. So, this project is based on a Text Mining technique of the tweets posted on Twitter and a sentiment analysis of them. We first search for the best machine learning algorithm and then extract from data some useful information that allows us to understand the position of Italian people with respect to this topic.

2 Dataset

2.1 Dataset Building

The dataset used in this application is composed of a great number of tweets retrieved by scraping all the tweets containing the word '*greenpass*' from the Twitter official website. The scraping consists in downloading the tweets from 2021-07-01 up to 2021-12-15.

The tweets have been obtained by using a Python script. The most relevant part of the code is shown in the picture below:

```
text_query = "greenpass"
since_date = "2021-06-01"
until_date = "2021-10-12"

# Using OS library to call CLI commands in Python
os.system('snscrape --jsonl --max-results {} --since {} twitter-search "{} until:{}"> text-query-tweets.json'.format(tweet_count, since_date, text_query, until_date))

# Reads the json generated from the CLI command above and creates a pandas dataframe
tweets_df2 = pd.read_json('text-query-tweets.json', lines=True)

# Displays first 5 entries from dataframe
# tweets_df2.head()

# Export dataframe into a CSV
tweets_df2.to_csv('text-query-tweets.csv', sep=',', index=False)
```

In this way, we collected around 1.068.130 tweets, with 28 attributes from 2021-07-01 up to 2021-12-15. Before doing any kind of analysis we discard answer tweets, tweets with not Italian content and moreover we delete all the duplicates. At the end our dataset is composed by 482.668 tweets with 28 attributes: _type, url, date, content, renderedContent, id, user, replyCount, retweetCount, likeCount, quoteCount, conversationId, lang, source, sourceUrl, sourceLabel, outlinks, tcooutlinks, media, retweetedTweet, quotedTweet, inReplyToTweetId, inReplyToUser, mentionedUsers, coordinates, place, hashtags, cashtags.

The distribution of the tweets in the different months is shown in the picture below:

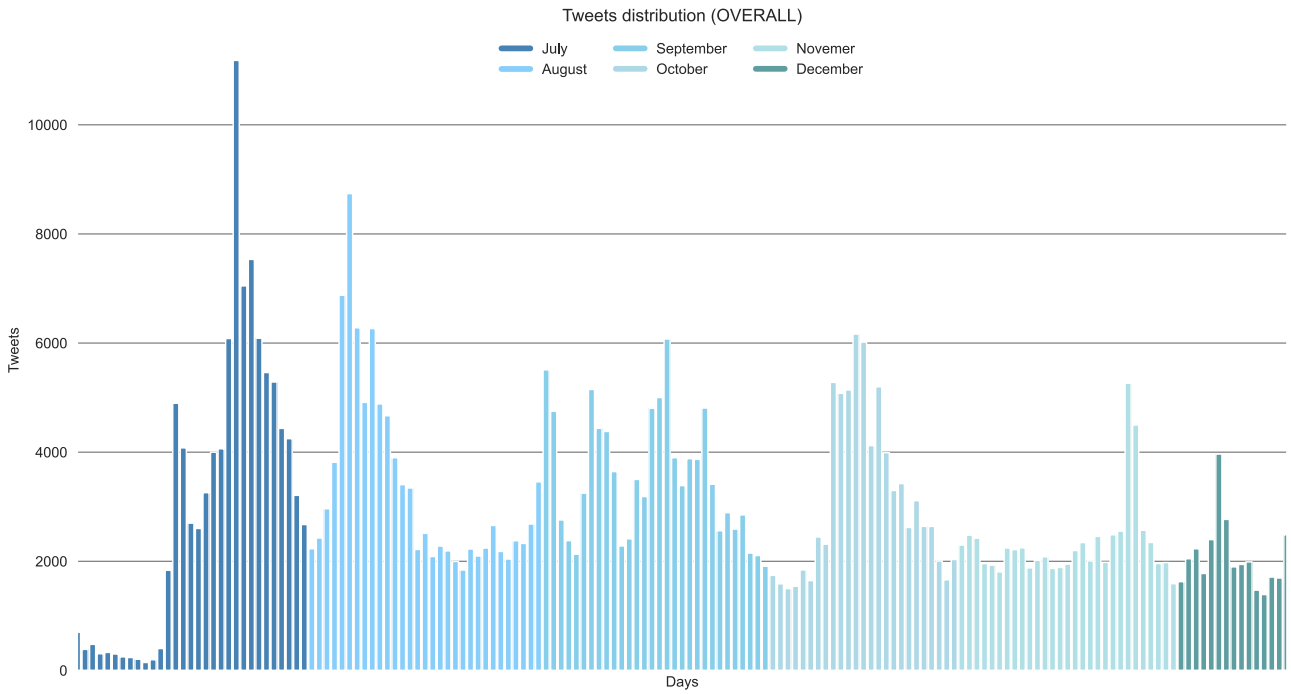
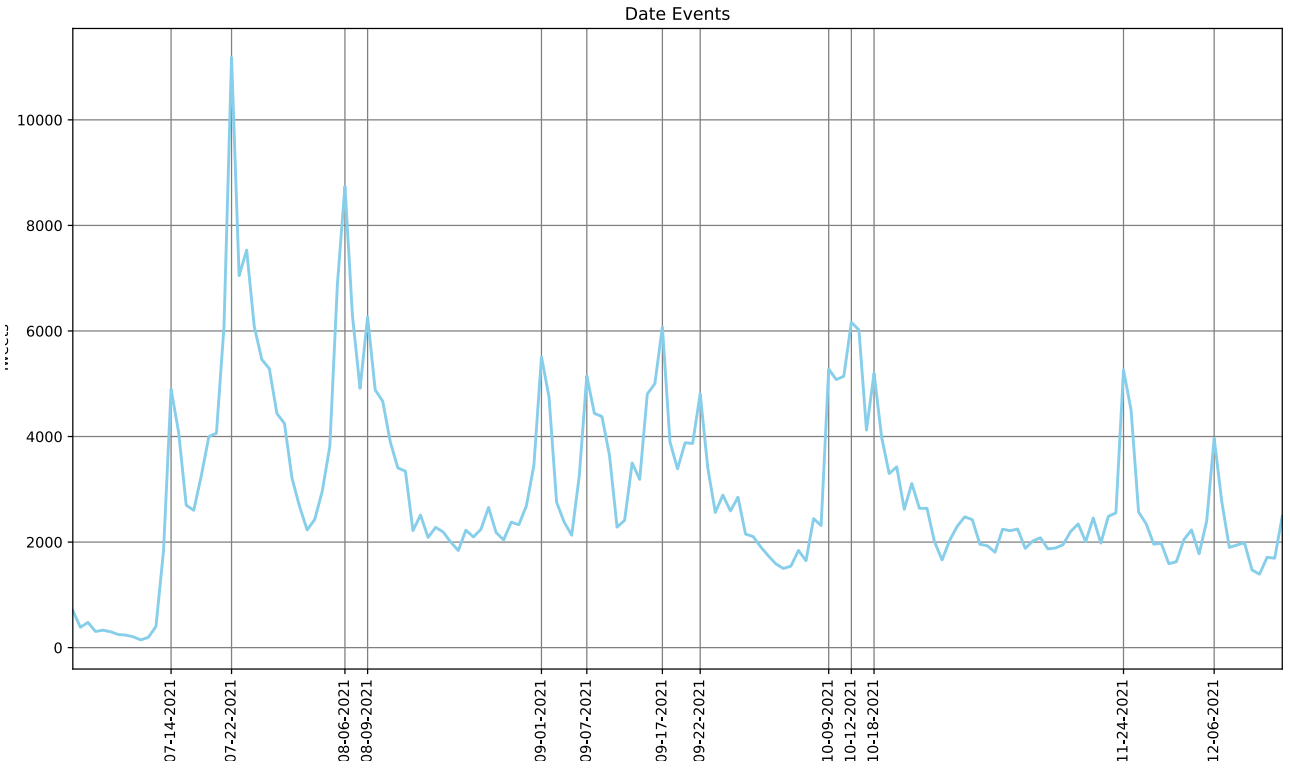


Figure 2.1 Tweets distribution from July to half of December



In the picture above we highlight the days in which the number of tweets generated by the users is higher than the other days. These special days represent special events that led people to express their opinion. The topic is the same, but the words used to describe the events could be different, new words could be used. The data may be expected to change over time. The following table shows the 13 peaks identified in the graph with the related probable reasons.

Event	Tweets	Description	Name
07-14-2021	4896	Italian government is starting to think about introducing the green pass on the French model.[1]	Peak_0
07-22-2021	11180	Council of Ministers approved the mandatory green pass to access bars, restaurants, cinemas, and theaters from 6 August.[2]	Peak_1
08-06-2021	8739	Green pass becomes mandatory in Italy.[3]	Peak_2
08-09-2021	6239	The interior minister clarifies that the owners of business cannot ask customers for documents.[4]	Peak_3

[1] https://www.rainews.it/dl/rainews/articoli/Green-Pass-Sileri-Seguire-subito-l-esempio-della-Francia-4b78fb78-5225-48d7-a326-293a9746d637.html

[2] https://www.ilfattoquotidiano.it/2021/07/22/nuovo-decreto-covid-dal-6-agosto-green-pass-per-ristoranti-al-chiuso-palestre-cinema-draghi-condizione-per-tenere-attivita-aperte/6270357/

[3]

[4] https://www.ilsole24ore.com/art/green-pass-lamorgesei-titolari-attivita-non-chiederanno-documenti-controlli-campione-AEcHEBc

08-09-2021	6265	Government specifies that the owners of the activities cannot ask for the documents, the checks on the green pass will be done randomly by the police.[5] Telegram channels were discovered selling fake green passes.[6]	Peak_4
09-01-2021	5510	The first no green pass events take place in the main Italian train stations, but they are not very successful.[7]	Peak_5
09-07-2021	5150	The Italian government is considering the hypothesis of extending the Green Pass obligation to public and private workers.[8]	Peak_6
09-17-2021	6074	Green Pass becomes mandatory for public employees: from 15 October until the end of the year.[9]	Peak_7
09-22-2021	4809	Public employees who do not have the green pass will not receive their salary from October 15th.[10]	Peak_8
10-09-2021	5279	Green Pass becomes mandatory to access schools and universities[11].	Peak_9
10-12-2021	6123	People react to the extension of the green pass in the workplace.[12]	Peak_10
10-18-2021	5198	Protests by workers at the port of Trieste.[13]	Peak_11
11-24-2021	5263	The Italian government introduces the *super green pass* from 6 December, that you can only receive with after getting vaccinated or after recovering from the disease.[14]	Peak_12
12-06-2021	3965	The super green pass becomes active.[15]	Peak_13

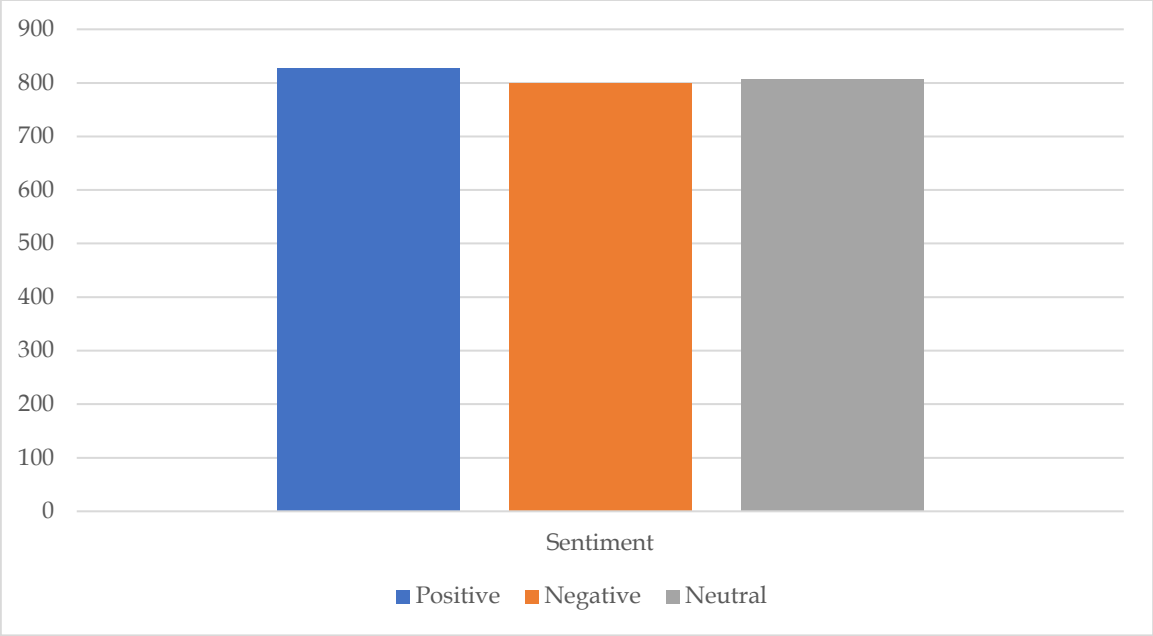
[5] https://www.ilsole24ore.com/art/green-pass-lamorgesei-titolari-attivita-non-chiederanno-documenti-controlli-campione-AEcHEBc?refresh_ce=1
[6] https://www.rainews.it/dl/rainews/articoli/Green-pass-vendita-di-falsi-certificati-sequestrati-32-canali-Telegram-28ed5be9-0150-4653-8e66-6d4cb3ad316a.html?refresh_ce
[7] https://www.ilfattoquotidiano.it/2021/09/01/no-green-pass-il-tentativo-di-bloccare-i-treni-e-un-flop-poche-decine-di-manifestanti-in-tutta-italia-si-presentano-fuori-dalle-stazioni/6307122/
[8] https://www.qualitytravel.it/estensione-obbligo-di-green-pass-per-i-lavoratori-gli-scenari-possibili-per-eventi-turismo-e-ristorazione/99667
[9] https://www.informazionefiscale.it/Green-Pass-obbligatorio-dipendenti-pubblici
[10] https://www.rainews.it/dl/rainews/articoli/green-pass-coronavirus-Dl-stipendio-f809dea2-8a59-4ecb-8050-a5bf87675578.html
[11] https://www.altalex.com/documents/news/2021/09/10/green-pass-nuove-regole-per-scuole-universita-e-rsa
[12] https://www.fanpage.it/politica/arriva-lapp-per-il-controllo-del-green-pass-a-lavoro-ecco-come-funzionera/
[13] https://www.ilfattoquotidiano.it/2021/10/18/no-green-pass-al-porto-di-trieste-polizia-sgombera-i-manifestanti-con-gli-idranti-il-video/6358427/
[14] https://www.governo.it/it/articolo/comunicato-stampa-del-consiglio-dei-ministri-n-48/18639
[15] https://tg24.sky.it/salute-e-benessere/2021/12/06/super-green-pass-download

2.2 Training set

For the training set, we built a dataset made up of 2400 tweets, labelled by hand. Each tweet belongs to one of the following classes:

- *Positive (1)*: tweet expressing a favorable opinion about green pass.
- *Negative (-1):* tweet expressing a contrary opinion about green pass.
- *Neutral (0):* tweet expressing neither a favorable opinion about green pass nor a negative one.

The training set has been built by randomly selecting an equal number of tweets for each class starting from 01-07-2021 up to 22-07-2021.



2.3 Dataset Pre-Processing

Raw tweets scraped from Twitter generally result in a noisy dataset. This is due to the casual nature of people's usage of social media. Tweets have certain special characteristics such as retweets, emoticons, user mentions, etc. which have to be suitably extracted. Therefore, raw twitter data must be normalized to create a dataset which can be easily learned by various classifiers. We have applied an extensive number of pre-processing steps to standardize the dataset and reduce its size. We first apply some general pre-processing techniques on tweets including:

- Converting the tweet to lower case.
- Replacing 2 or more dots (.) with space.
- Stripping spaces and quotes (" and ') from the ends of tweet.
- Replacing 2 or more spaces with a single space.

Moreover, regular expressions were exploited in order to remove user mentions, URLs, emoticons, retweets. In this way, at the end we obtained a dataset composed of about 500.000 cleaned tweets.

In addition, punctuation marks, brackets, quotes, special characters have been removed.

2.4 Features Extraction

Before creating the model we need to apply a text mining process in order to obtain, from a set of strings, a set of numeric vectors that will be elaborated in the classification step, so we can represent a sentence as a string of numbers.

1. **Tokenization**: in this first step we transformed our text into a stream of processing units called tokens. In this way each text is represented as a set of words and for removing redundancy between tokens, they were converted in lowercase.
2. **Stop-word filtering**: in this second step we removed the stop words. Stop words are those words that are useless for the sentiment analysis such as conjunctions, prepositions or articles.
3. **Stemming**: in this last step we transformed our token into the relative stem in order to group words that have a similar semantic.
4. **Feature Extraction**: as last step we compute the TFIDF that reflects how important a word is to a document in a collection or corpus.
    a. The tf-idf score serves as a weight for each word and signifies the importance of each word in an instance, so helps to adjust for the fact that some words appear more frequently in general.

3 Model tuning

GridSearchCV is a model selection step, and this should be done after Data Processing tasks. It is always good to compare the performances of Tuned and Untuned Models in order to obtain the best results.

Hyperparameters for a model can be chosen using several techniques such as Random Search, Grid Search, Manual Search, Bayesian Optimizations, etc. In this project, we perform the GridSearchCV which uses the Grid Search technique for finding the optimal hyperparameters to increase the model performance.

Grid Search uses a different combination of all the specified hyperparameters and values. It calculates the performance for each combination and selects the best value for the hyperparameters. This makes the processing time-consuming and expensive based on the number of hyperparameters involved.

Cross-Validation is used while training the model. As we know, before training the model with data, we divide the data into two parts – train data and test data. In cross-validation, the process divides the train data further into two parts – the train data and the validation data.

Each iteration keeps one partition for testing and the remaining k-1 partitions for training the model. The next iteration will set the next partition as test data and the remaining k-1 as train data and so on. In each iteration, it will record the performance of the model and at the end gives the average of all the performances.

For this reason, we did not perform the test for all the possible combinations, due to the very long time required to do so.

Primarily, it takes 4 arguments i.e., *estimator*, *param_grid*, *cv,* and *scoring*. The description of the arguments is as follows:

1. *estimator* – A scikit-learn model

2. *param_grid* – A dictionary with parameter names as keys and lists of parameter values.

3. *scoring* – The performance measure. For example, 'r2' for regression models, 'precision' for classification models.

4. *cv* – An integer that is the number of folds for K-fold cross-validation.

Thus, clf.best_params_ gives the best combination of tuned hyperparameters, and clf.best_score_ gives the average cross-validated score of our classifier.

3.1 Baseline Tuning

SVM	**TF-IDF + Uni**	Range
Max_df	0.65	0.65, 0.75, 0.85, 1.0
k	3000	'all', 1000, 2000, 2500, 3000, 3500, 3700
C	10	0.01, 0.1, 1, 10, 100
gamma	1	1, 0.1, 0.01, 0.001, 0.0001
kernel	rbf	rbf, linear

SVM	**Accuracy(%)**		**Precision(%)**		**Recall(%)**		**F1 – score(%)**		**Exe Time (Min)**
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
TF-IDF + Uni	65,41	0,0271	65,42	0,0325	65,42	0,0270	65,43	0,0371	**54**

Log Reg	**TF-IDF + Uni**	Range
max_df	0.65	0.65, 0.75, 0.85, 1.0
k	3500	'all', 1000, 2000, 2500, 3000, 3500, 3700
C	1	0.01, 0.1, 1, 10, 100
max_iter	1500	1500

Logistic Regression	**Accuracy(%)**		**Precision(%)**		**Recall(%)**		**F1 – score(%)**		**Exe Time (Min)**
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
TF-IDF + Uni	64,06	0,0342	64,05	0,0342	64,08	0,0342	64,06	0,0342	**2**

3.2 Naïve bayes Tuning

Multi	**TF-IDF + Uni**	Range
Max_df	0.65	0.65, 0.75, 0.85, 1.0
k	3500	'all', 1000, 2000, 2500, 3000, 3500, 4000
alpha	1	1, 1e-1, 1e-2

Multinomial	**Accuracy(%)**		**Precision(%)**		**Recall(%)**		**F1 – score(%)**		**Exe Time (Min)**
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
TF-IDF + Uni	64,92	0,0284	64,85	0,0332	65,08	0,0283	64,95	0,0283	**<1**

Compl	**TF-IDF + Uni**	Range
Max_df	0.65	0.65, 0.75, 0.85, 1.0
k	3500	'all', 1000, 2000, 2500, 3000, 3500, 4000
alpha	1	1, 1e-1, 1e-2

Complement	**Accuracy(%)**		**Precision(%)**		**Recall(%)**		**F1 – score(%)**		**Exe Time (Min)**
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
TF-IDF + Uni	65,46	0,030	65,39	0,030	63,58	0,0301	65,49	0,0301	**<1**

3.3 Ensemble Tuning

Random Forest	**Uni**	Range
Max_df	0,85	0.65, 0.75, 0.85, 1.0
k	4000	'all', 1000, 2000, 2500, 3000, 3500, 4000
criterion	gini	gini, entropy
min_samples_split	10	2, 5, 10
n_estimators	100	100, 300, 500, 750, 800, 1200

Random Forest	**Accuracy(%)**		**Precision(%)**		**Recall(%)**		**F1 – score(%)**		**Exe Time (Min)**
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
Uni	62,38	0,0317	62,35	0,0294	62,47	0,0316	62,41	0,0294	**205**

Adaboost	**TF-IDF + Uni**	Range
Max_df	0.65	0.65, 0.75, 0.85, 1.0
k	2000	'all', 1000, 2000, 2500, 3000, 3500, 4000
Learning_rate	0.1	0.001, 0.01, 0.1, 0.2, 0.5
N_estimators	1500	100, 500, 1000, 1500

Adaboost	**Accuracy(%)**		**Precision(%)**		**Recall(%)**		**F1 – score(%)**		**Exe Time (Min)**
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
TF-IDF + Uni	57,57	0,0384	57,50	0,0382	57,58	0,0384	57,59	0,0383	**72**

Gradient Boosting	**TF-IDF + Uni**	Range
Max_df	1.0	0.65, 0.75, 0.85, 1.0
k	4000	'all', 1000, 2000, 2500, 3000, 3500, 4000
Learning_rate	0.2	0.001, 0.01, 0.1, 0.2, 0.5
N_estimators	500	100, 500, 1000, 1500

Gradient Boosting	**Accuracy(%)**		**Precision(%)**		**Recall(%)**		**F-score(%)**		**Exe Time (min)**
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
TF-IDF + Uni	61,72	0,0244	61,71	0,0245	61,74	0,0246	61,74	0,0245	363

Bag + SVM	**TF-IDF + Uni**	Range
Max_df	1.0	0.65, 0.75, 0.85, 1.0
k	All	'all', 1000, 2000, 2500, 3000, 3500, 4000
N_estimators	10	10, 30, 50, 100

Bagging + SVM	**Accuracy(%)**		**Precision(%)**		**Recall(%)**		**F-score(%)**		**Exe Time (min)**
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
TF-IDF + Uni	64,72	0,0227	64,73	0,0230	64,74	0,0201	64,73	0,0203	50

Bag + SVM	**TF-IDF + Uni**	Range
Max_df	0.85	0.65, 0.75, 0.85, 1.0
k	3000	'all', 1000, 2000, 2500, 3000, 3500, 4000
N_estimators	100	10, 30, 50, 100

Bagging + Log Reg	Accuracy(%)		Precision(%)		Recall(%)		F1 – score(%)		Exe Time (Min)
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
TF-IDF + Uni	64,96	0,0315	64,96	0,0315	64,99	0,0317	64,97	0,0316	**27**

After this step we discard some combinations and consider the best parameters to build the final model, but before we have to evaluate the performance for each best combination of each classifier.

4 Model Performance

In this section it is reported the performance class by class of the classifier trained with the best parameters selected previously.

In the table are described the performances for each class measuring the performance using '*precision*', '*recall*' and '*f1*' score, and of course the '*accuracy*'.

We want to select the classifier that better predicts tweets belonging to the negative and positive classes. In general, we have achieved an accuracy between 64% and 65% on the training set.

4.1 Compare SVM with Logistic Regression by classes:

Classifier	Accuracy (%)	Precision (%) by class			Recall(%) by class			F-score(%) by class		
		Pos	Neg	Neu	Pos	Neg	Neu	Pos	Neg	Neu
SVM	64,06	66,00	67,14	59,11	68,47	64,12	59,47	67,21	65,60	59,29
Logistic Regression	64,88	66,12	67,90	60,60	68,59	67,00	59,85	67,33	64,88	60,22

Both the algorithms perform quite well in positive and negative classes, they are worse in neutral tweets prediction. Moreover, we can say that the algorithms perform quite well on the dataset, in fact the results are quite similar, it seems that SVM is better in recognizing negative class while logistic regression is better in recognizing positives class, but the difference is really small, indeed we proved the similarity in performances of the two algorithms with a statistical test.

4.2 Compare MultinomialNB with ComplementNB:

Classifier	Accuracy (%)	Precision (%) by class			Recall(%) by class			F-score(%) by class		
		Pos	Neg	Neu	Pos	Neg	Neu	Pos	Neg	Neu
MultinomialNB	64,43	58,08	74,95	65,95	82,00	57,25	53,53	68,00	64,91	59,09
ComplementNB	65,21	60,55	72,42	65,27	78,62	62,37	54,27	68,41	67,02	59,26

Both the algorithms perform quite well in positive and negative classes, they are slightly worse in neutral tweets prediction. Moreover, we can say that the algorithms perform quite well on the

dataset, in fact the results are quite similar, even though ComplementNB is slightly better in recognizing both positives and negatives tweets.

4.3 Compare Bagging + SVM with Bagging + Logistic Regression:

Classifier	Accuracy (%)	Precision (%) by class			Recall(%) by class			F-score(%) by class		
		Pos	Neg	Neu	Pos	Neg	Neu	Pos	Neg	Neu
Bagging + SVM	63,86	67,28	65,31	59,01	65,57	66,37	59,60	66,42	65,84	59,30
Bagging + Log Reg	64,59	66,08	67,90	59,80	68,23	66,12	59,35	67,14	67,00	59,57

Both the algorithms perform quite well in positive and negative classes, they are slightly worse in neutral tweets prediction. Moreover, we can say that the algorithms perform quite well on the dataset, in fact the results are quite similar, even though Bagging + Logistic Regression is slightly better in recognizing both positives and negatives tweets, but the difference is really small, indeed we proved the similarity in performances with a statistical test.

In general, all the classifiers – but *random forest* and *gradient boosting* - perform quite well in the prediction of positive and negative tweets.

Now, we compare the classifiers 2 by 2 in order to reduce the list.

In this way, we further reduce the list of the candidate classifiers. According to the chosen metrics, the best classifier seems to be ComplementNB since it returns better results than all the other classifiers.

4.4 Paired t-test

Once the models have been generated, it is important to choose the most suitable classifier for our task. It is not enough to use average accuracy as a comparison variable, it is instead necessary to use a statistical approach that highlights if there is any real and significant difference in the accuracy of two models.

A t-test is used to compare and determine whether the mean difference between two sets of observations is zero. The test started with the null hypothesis, in which we propose that no significant difference exists in a set of given distribution, i.e., we assume that the distribution is a normal distribution and so the difference in mean error rate between the classifiers is zero.

This test-statistic is then compared with a critical value and if it is found to be greater than the critical value the null hypothesis is rejected, stating that the two models are statistically different.

The t-test computes the t-statistic with k -1 degrees of freedom for k samples.

1. Calculate test statistic

The t-score is the ratio between the difference between two groups and the difference within the groups. The larger the t score, the more difference there is between groups. The smaller the t score, the more similarity there is between groups.

$$t = \frac{x_1 - x_2}{\sqrt{\frac{\sigma}{N}}}$$

where x_1 and x_2 are the two sets, σ is the variance of the difference of the two sets and N is the number of samples.

2. Select significance level (p = 5%)

We performed the t-test using a significance level equals to 0.05. The benefit of using p-value is that it calculates a probability estimation, we can test at any desired level of significance by comparing this probability directly with the significance level.

The p-value is the probability that the results from your sample data occurred by chance. It is preferable to have low values of p. In fact, low values of p indicate that your data did not occur by chance.

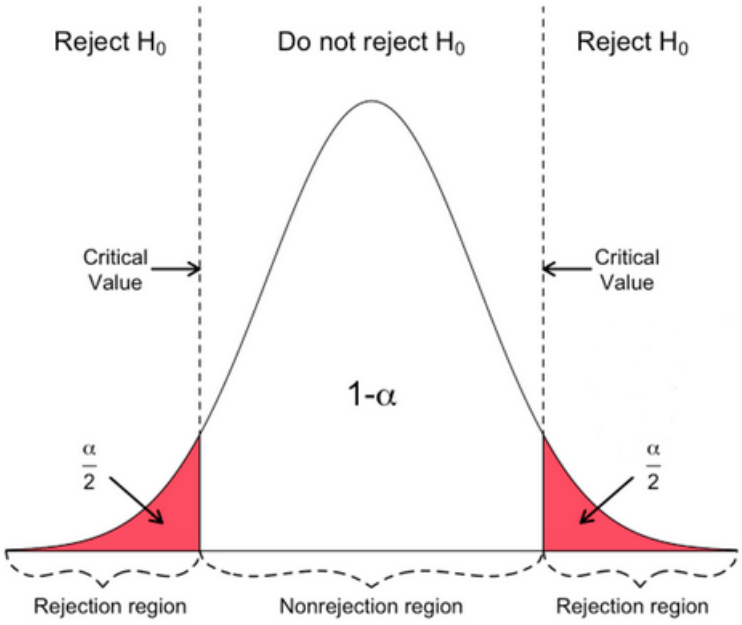
3. Consult table for t distribution: Find t-value corresponding to k-1 degrees of freedom and based on significance level

In this case k is equal to the number of cross-validation samples used.

4. Compare test statistic with critical values

A critical value is a point (or points) on the scale of the test statistic beyond which we reject the null hypothesis, and it is derived from the level of significance α of the test. Critical value can tell us what is the probability that two samples belong to the same distribution. The higher the critical value, the lower the probability of two samples belonging to same distribution.

If the computed t is inside the critical region, we can reject the null-hypothesis.



If null hypothesis is rejected, it suggests that the difference in skill scores is statistically significant.

So, for each model, we performed the 10-fold cross-validation, that is 10 times, each time using a different 10-fold data partitioning. Each partitioning is independently drawn. We can average the 10 obtained accuracies each for M1 and M2, respectively, to obtain the mean accuracy for each model.

In general, they follow a t-distribution with k-1 degrees of freedom where, here, k = 10. This distribution looks very similar to a normal, or Gaussian, distribution even though the functions defining the two are quite different. Both are unimodal, symmetric, and bell-shaped.

We performed the t-test between each classifier and all the others in order to understand if two models perform equally well on the dataset. We started from the null hypothesis that the two distributions of accuracy for the 2 models are the same. This test will help us if we can accept or reject the null hypothesis.

VS	LOG REG	MULTI_NB	COMP_NB	BAG_SVM	BAG_LOG
SVM	4.3883	1.5077	-1.2245	7.6961	7.7386
	0.6467 / 0.6522	0.6494 / 0.6522	0.6545 / 0.6522	0.6456 / 0.6522	0.6444 / 0.6522
LOG REG		-1.5650	-6.0565	0.8210	2.6656
		0.6494 / 0.6467	0.6545 / 0.6467	0.6456 / 0.6467	0.6444 / 0.6467
MULTI_NB			-3.459	1.7161	2.2671
			0.6545 / 0.6494	0.6456 / 0.6494	0.6444 / 0.6494
COMP_NB				4.3025	5.5733
				0.6456 / 0.6545	0.6444 / 0.6545
BAG_SVM					1.0879
					0.6444 / 0.6456

Table 4.1 t-score result and relative accuracy (10 x 10-Cross Validation)

Green: We cannot reject the null hypothesis and may conclude that the performance of the two algorithms is not significantly different.

Red: We can reject the null hypothesis that both models perform equally well on this dataset. We may conclude that the two algorithms are significantly different.

From this table we can see that there are 6 cases in which there are no significant differences between classifiers, in fact the t-score value lies inside the *Nonrejected region*.

Starting from SVM, it is statistically similar to MultinomialNB and ComplementNB. So, we choose ComplementNB because it is the one with the highest accuracy.

Logistic Regression is statistically similar to MultinomialNB and Bagging + SVM. So, we choose MultinomialNB because it is the one with the highest accuracy.

MultinomialNB is statistically similar to BaggingSVM, LogisticRegression and SVM. So, we choose SVM because it is the one with the highest accuracy.

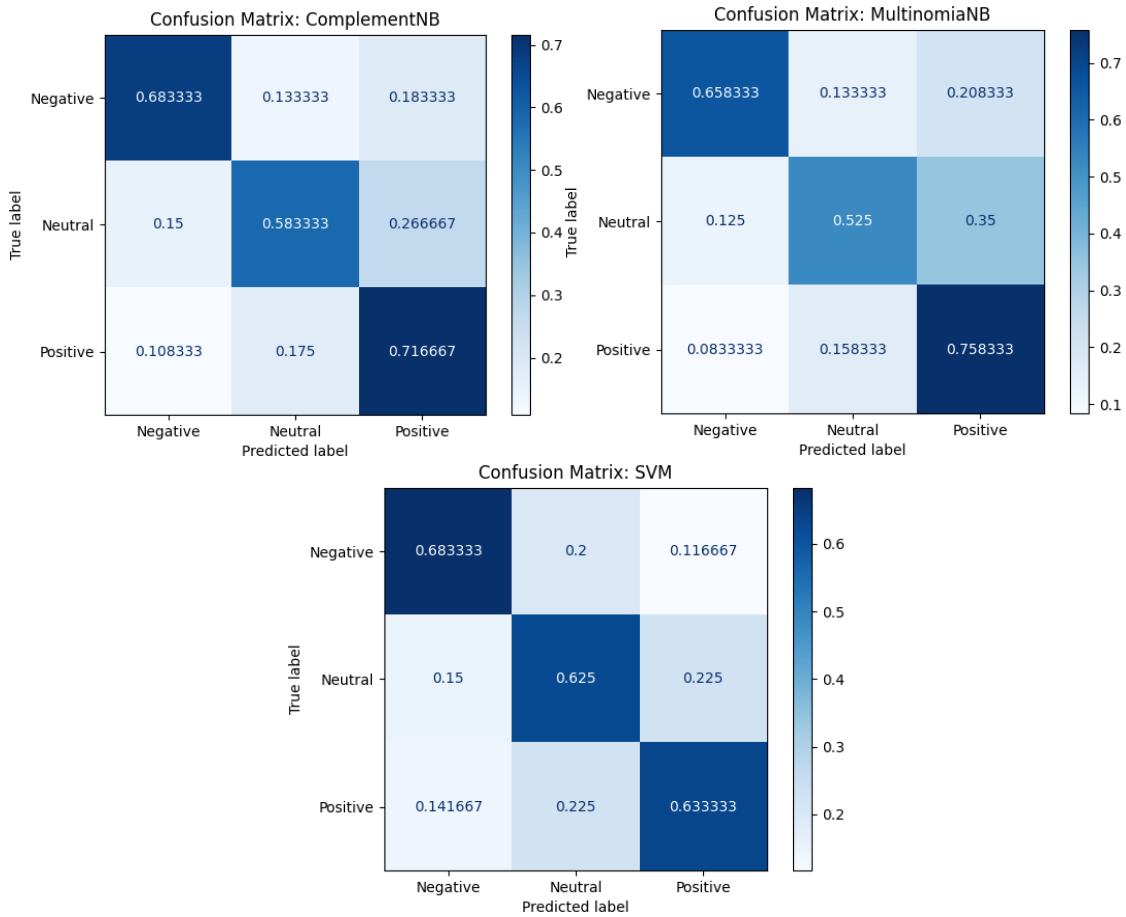
Bagging + SVM is similar to Bagging + Logistic regression, Logistic regression and MultinomialNB. So, we choose MultinomialNB because it is the one with the highest accuracy.

5 Model Selection

Finally, we try to understand which classifier performs better in recognizing tweets that comes from another peak. We labeled 360 tweets (with equally distributed classes) from the 23rd of July up to 6th of August, and we test the performance on the tree best models.

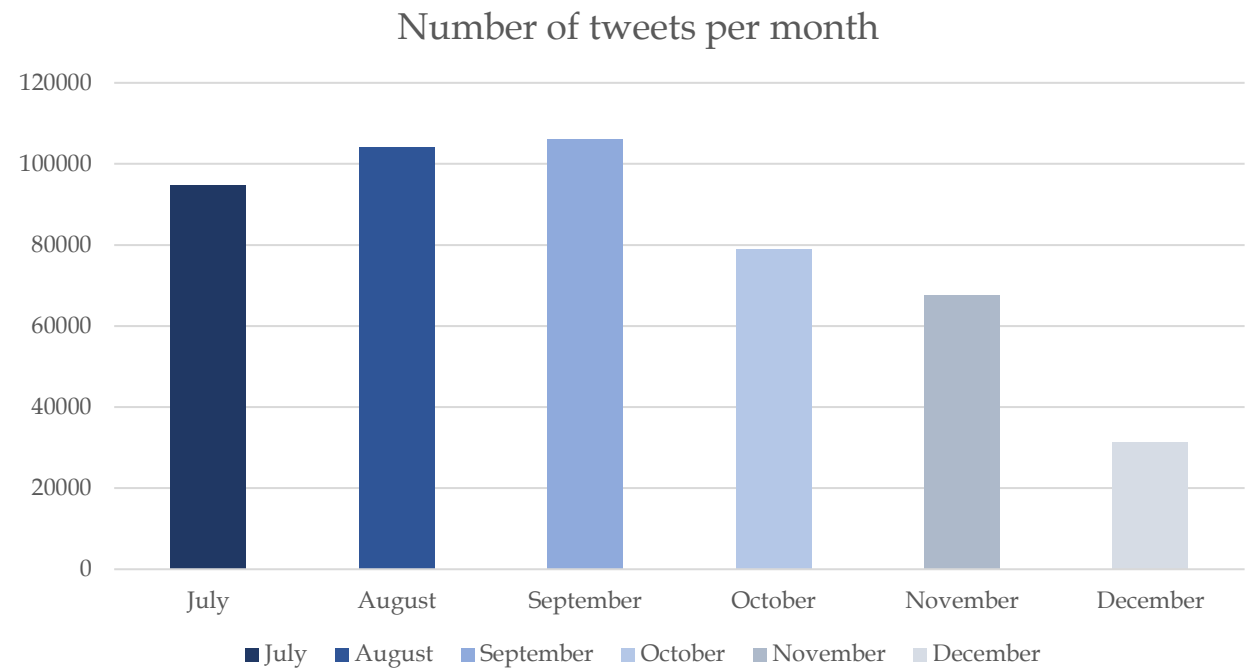
Classifier	**Accuracy (%)**	Precision (%) by class			Recall(%) by class			F-score(%) by class		
		Pos	Neg	Neu	Pos	Neg	Neu	Pos	Neg	Neu
ComplementNB	66,11	61,42	0.7256	65,42	71,66	68,33	58,33	66,15	70,38	61,67
MultinomialNB	64,72	57,59	75,96	64,28	75,83	65,83	52,50	65,46	70,53	57,79
SVM	64,72	64,95	70,08	59,52	63,33	68,33	62,50	64,13	69,19	60,97

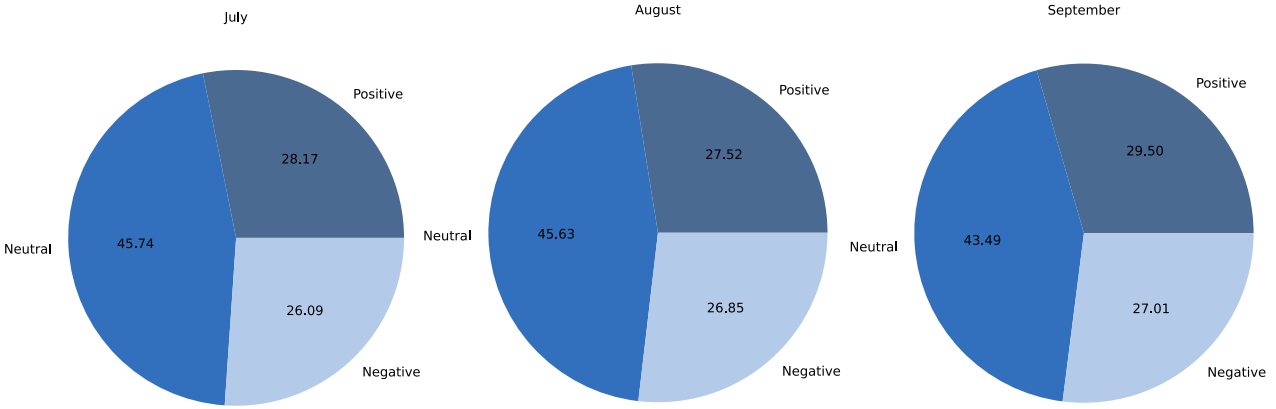
We can conclude that the best model for our application domain is *ComplementNB*, so we continue our study by using this model.

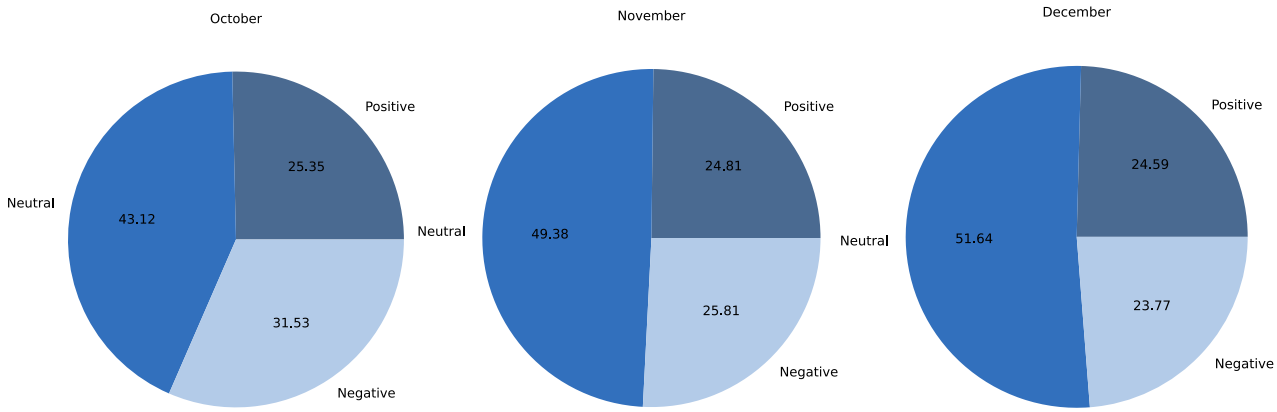


5.1 Distribution of opinion polarity over the classes by month using the selected model

After selecting the model, we performed a monthly analysis. The figures below, in fact, show the number of tweets per month and the distribution of tweets over the three classes.







Neutral tweets are the most notable, both in terms of frequency and of amplitude. However, neutral tweets, for the most part, correspond to news tweets and indicate users talking about green pass or sharing news in correspondence with the event, whereas sometimes correspond also to personal objective texts without a clear opinion.

6 Monitoring Analysis

With the learning techniques we use these days, a model is never final. In training, it studies the past examples. Once released into the wild, it works with new data. With time, this data deviates from what the model has seen in training. Sooner or later, even the most accurate and carefully tested solution starts to degrade.

6.1 Concept drift

When dealing with continuous classification of data streams along the time, the issue of concept drift should be analyzed. Indeed, the classification models are usually trained using data extracted in a specific time interval. Then, such models are used for classifying the new instances received in streaming. Since the characteristics of the phenomenon under observation can change with time, the performance of the classification models may deteriorate, due to this concept drift. Thus, once in the classification system the presence of concept drift is detected, appropriate strategies for reducing it have to be possibly applied. In the context analyzed in this work, in which we carry out a classification of stance about greenpass in Italy from tweets, users of Twitter may change over time the words and/or phrases used for expressing their opinion. For this reason, we decided to carry out an additional experimental analysis for detecting the presence of concept drift along the time span under observation.

6.2 Study

The problem of the concept drift is present during the period analyzed in this project since the guidelines regarding green pass are constantly updated with the new rules and changes and so people react to those changes producing new tweets in which they express their opinions.

For these reasons we selected some key events present in our timespan, one for each month, and we performed a retrain of the model to cope with the problem of concept drift.

The events selected are:

Event	Tweets	Description	Name	New name
07-22-2021	11180	Council of Ministers approved the mandatory green pass to access bars, restaurants, cinemas, and theaters from 6 August.	Peak_1	Peak_1
08-06-2021	8739	Green pass becomes mandatory in Italy.	Peak_2	Peak_2
09-17-2021	6074	Green Pass becomes mandatory for public employees: from 15 October until the end of the year.	Peak_7	Peak_3

10-09-2021	5279	Green Pass becomes mandatory to access schools and universities.	Peak_9	Peak_4
11-24-2021	5263	The Italian government introduces the *super green pass* from 6 December, that you can only receive with after getting vaccinated or after recovering from the disease.	Peak_12	Peak_5
12-06-2021	3965	The super green pass becomes active.	Peak_13	Peak_6

Different solutions were compared for coping with the problem of handling concept drift in the classification of Twitter streams, so we perform a comparative study based on different schemas.

For each event selected we labelled by hand 360 tweets, excepts for peak #5 and #6, in which we have labeled 120 tweets, because the number of tweets decreased. We use those tweets as test set for 3 different learning settings:

- **Static model**: we consider only the initial training set composed by 2400 tweets used to generate the first ComplementNB classifier.
- **Sliding model**: retrained each time the naïve bayes classifier with the most recent tweets, removing the oldest 360 (120) and adding the newest 360 (120).
- **Incremental model**: retrained each time the model with an incremental training set, this means that we consider the previous training set (build at peak i-1, so considering all the previous events) adding new labelled data.

To detect the presence of drift, we evaluated the *F-measure*, the *precision*, the *recall* per class and the overall *accuracy*, obtained classifying the labelled tweets of each selected peak.

Before evaluating the classification performances on a specific event, we re-trained the classification models modifying the training set. With the static schema, instead, we always use the same model, trained at the beginning, to classify all the tweets from July 1st up to December 15th.

Incremental Model	**Accuracy(%)**		**Precision(%)**		**Recall(%)**		**F1 – score(%)**		**Number of features**
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
Peak 1	68,88	0.039	66,84	0.039	66,96	0.039	66,90	0.039	**5659**
Peak 2	68,22	0.049	68,19	0.049	68,28	0.049	68,24	0.049	**5995**
Peak 3	67,86	0.049	67,83	0.049	67,91	0.049	67,87	0.049	**6378**
Peak 4	68,46	0.052	68,44	0.052	68,50	0.052	68,47	0.052	**6699**
Peak 5	67,35	0.044	67,33	0.044	67,38	0.043	67,35	0.044	**6938**
Peak 6	67,53	0.052	67,51	0.052	67,56	0.052	67,53	0.052	**7064**

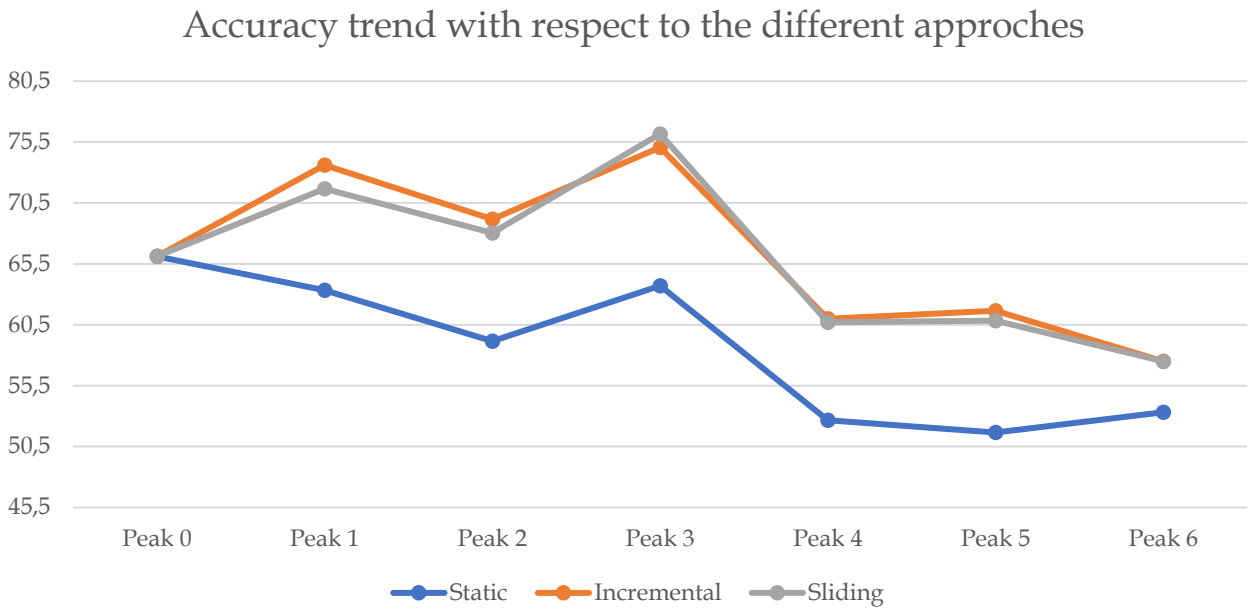
Sliding Model	Accuracy(%)		Precision(%)		Recall(%)		F1 – score(%)		Number of features
	Mean	Std dev	Mean	Std dev	Mean	Std dev	Mean	Std dev	
Peak 1	66,76	0.031	66,72	0.030	67,15	0.029	66,93	0.030	**5304**
Peak 2	67,53	0.050	67,52	0.050	67,82	0.048	67,67	0.049	**5329**
Peak 3	66,95	0.066	66,99	0.067	66,94	0.066	66,97	0.066	**3500**
Peak 4	68,41	0.0643	68,40	0.064	68,45	0.064	68,43	0.064	**5407**
Peak 5	68,59	0.059	68,54	0.059	68,70	0.058	68,62	0.059	**5363**
Peak 6	68,11	0.062	68,08	0.062	68,16	0.062	68,12	0.062	**4000**

6.3 Assessment of the results of the three approaches

Incremental Model	Accuracy (%)	Precision (%) by class			Recall(%) by class			F-score(%) by class		
		Pos	Neg	Neu	Pos	Neg	Neu	Pos	Neg	Neu
Peak 1	73,61	71,12	78,09	72,56	84,16	68,33	68,33	77,09	72,88	70,38
Peak 2	69,16	63,15	71,42	74,25	70,00	75,00	62,50	66,40	73,17	67,87
Peak 3	75,06	70,13	71,68	85,57	84,87	67,50	72,95	76,80	69,52	78,76
Peak 4	61,01	57,55	61,46	65,90	71,42	59,82	51,78	63,74	60,63	58,00
Peak 5	61,66	53,12	65,38	76,66	85,00	42,50	57,50	65,38	51,51	65,71
Peak 6	57,50	51,11	63,41	58,82	57,50	65,00	50,00	54,11	64,19	54,05
Average	**66,33**	**61,03**	**68,57**	**72,29**	**75,49**	**63,03**	**60,51**	**67,25**	**65,32**	**65,8**

Sliding Model	Accuracy (%)	Precision (%) by class			Recall(%) by class			F-score(%) by class		
		Pos	Neg	Neu	Pos	Neg	Neu	Pos	Neg	Neu
Peak 1	71,66	67,56	76,92	72,22	83,33	66,66	65,00	74,62	71,42	68,42
Peak 2	68,05	63,70	72,50	68,57	71,66	72,50	60,00	67,45	72,50	64,00
Peak 3	76,17	73,33	72,17	83,78	83,19	69,16	76,22	77,95	70,63	79,82
Peak 4	60,71	56,52	58,47	71,25	69,64	61,60	50,89	62,39	60,00	59,37
Peak 5	60,83	53,22	63,63	76,00	82,50	52,50	47,50	64,70	57,53	58,46
Peak 6	57,50	50,00	57,77	66,66	52,50	65,00	55,00	51,21	61,17	60,27
Average	**65,82**	**60,72**	**66,91**	**73,08**	**73,8**	**64,57**	**59,1**	**66,39**	**65,54**	**65,06**

Static Model	Accuracy (%)	Precision (%) by class			Recall(%) by class			F-score(%) by class		
		Pos	Neg	Neu	Pos	Neg	Neu	Pos	Neg	Neu
Peak 1	63,33	61,65	71,56	58,40	68,33	60,83	60,83	64,82	65,76	59,59
Peak 2	59,16	50,70	70,19	59,64	60,00	60,83	56,66	54,96	65,17	58,11
Peak 3	63,71	60,13	66,34	66,05	74,78	57,50	59,01	66,66	61,60	62,33
Peak 4	52,67	51,19	60,25	48,88	76,78	41,96	39,28	61,42	49,47	43,56
Peak 5	51,66	46,55	56,52	56,41	67,50	32,50	55,00	55,10	41,26	55,69
Peak 6	53,33	45,00	64,86	51,16	45,00	60,00	55,00	45,00	62,33	53,01
Average	**57,31**	**52,54**	**64,95**	**56,76**	**65,4**	**52,27**	**59,1**	**57,99**	**57,6**	**55,38**



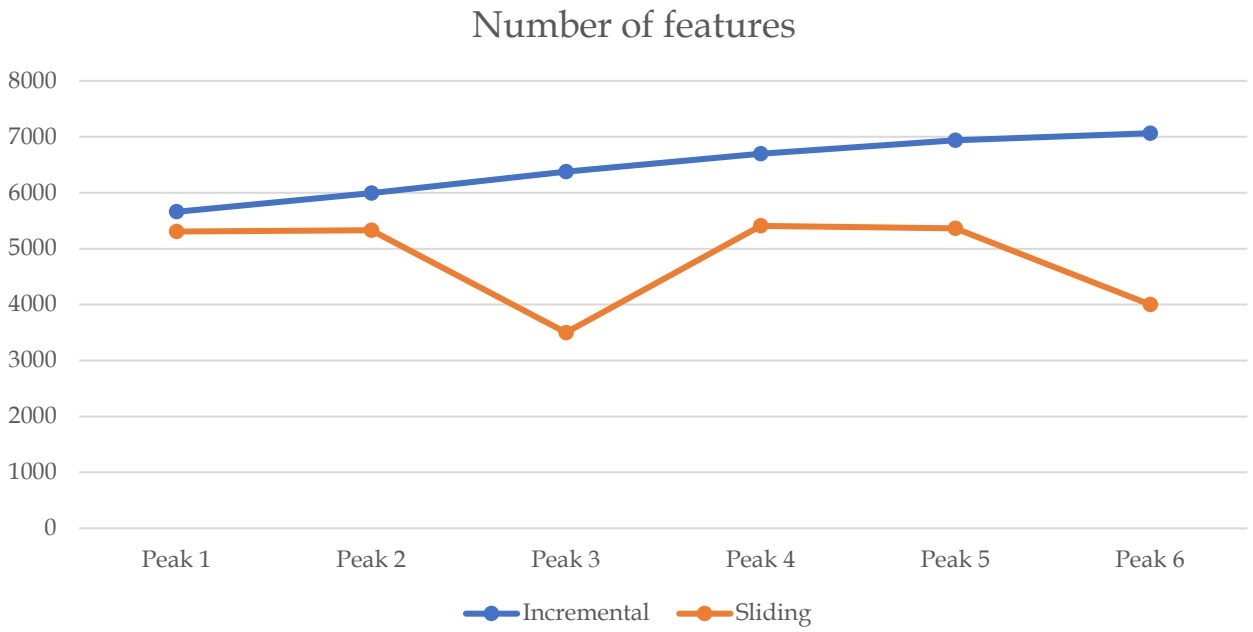
The graph above shows a plot of the trends of the accuracy over the time span of the selected events for the three classification models. In the figure, the continuous blue line, the continuous orange line, and the continuous grey line show the trend of the accuracy achieved, respectively, by the static model, the incremental model, and the sliding model.

We can note that the incremental and sliding solutions outperforms the static model scheme in each peak. While both sliding and incremental models achieve very similar results in each peak, even if the first one is slightly better: at peak 1, 2, 4 and 5 incremental overcomes sliding accuracy, while only in pick 3 sliding performs better. But in general, the performance of the two models is quite similar.

On the other hand, something strange happens after peak 4, in the worst case the accuracy of incremental and sliding models starts to deteriorate down to a value below 60%. Even in the static model the performances decrease drastically, achieving an accuracy close to 50% in the worst case.

Thus, we can affirm that sliding and incremental approaches are preferable to a static learning. So, incremental and time windows learning are a better approach when we deal with concept drift on tweets.

Another consideration must be done on the dictionary size. Although the static model uses the same dictionary every time, the sliding and incremental models update their dictionaries during the time. The sliding model changes its dictionary removing the oldest tweets and adding the newest one, instead the incremental model adds always new tweets increasing the size of the dictionary as shown in the plot below.



Some considerations must be done on incremental and sliding dictionary because the two schemes behave quite well even if incremental has slightly better results. However, it is necessary to analyze the number of features used by the two models.

In the incremental approach the number of features used for the training of the model grows more and more, peak by peak, reaching up to 7000 features, while in the case of sliding this does not always happen, and in general a smaller number of features are required.

In the end, we decided to select the *sliding* approach to conclude our analysis because, on average, it gave us results that are comparable (even slightly better sometimes) with the incremental approach, and it requires less features.

6.4 Results

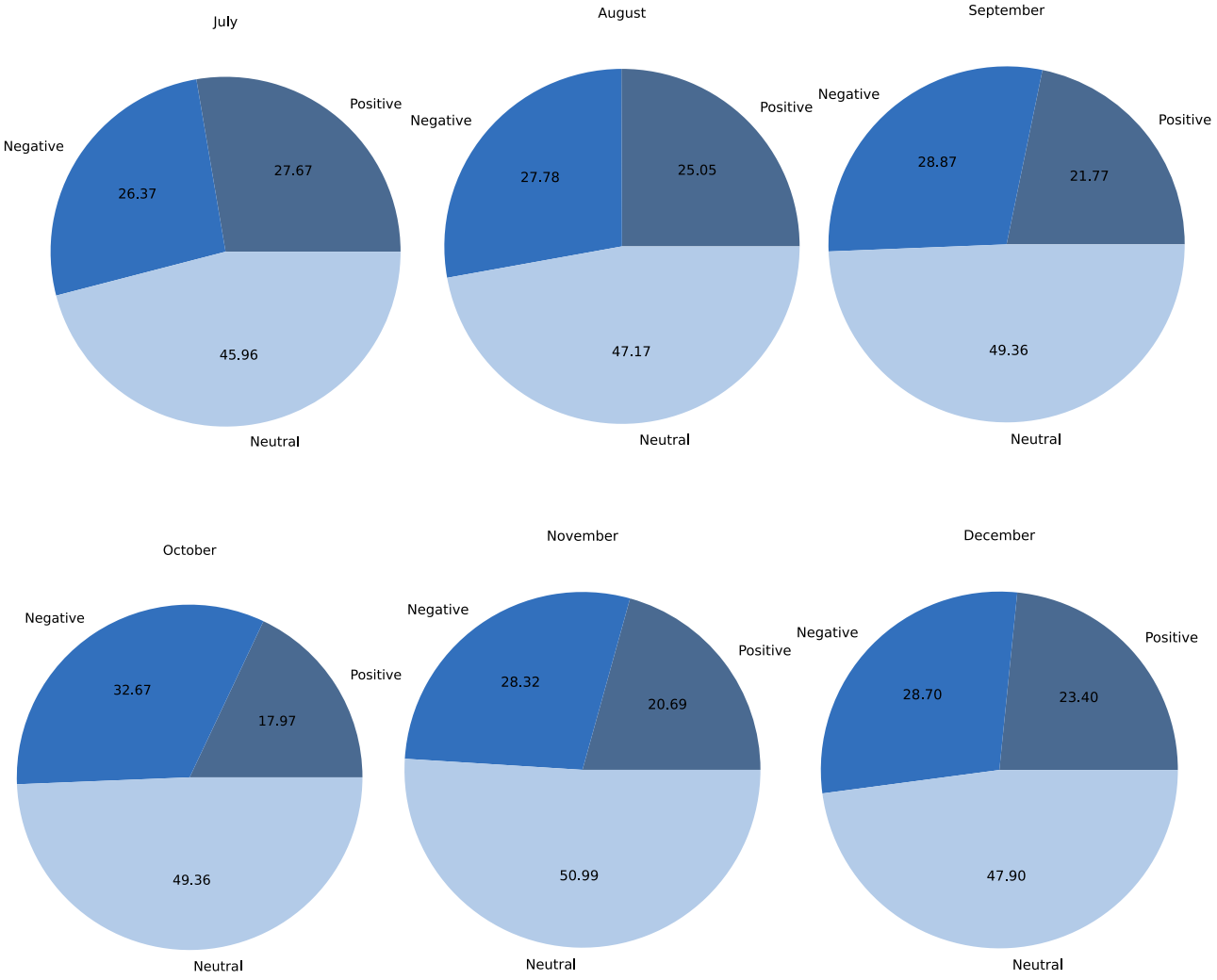
The analysis performed gave us some interesting results, although it was just related to a short period of time. As we can see from the image of the accuracies, since the beginning we notice concept drift, indeed the accuracy of the static model immediately starts decreasing.

This may be due to the fact that at each event new areas are added by the Italian government in which the green pass has to be exhibited, so the topics on which users mainly focus, change after each peak (sometimes even within the same peak). Until peak #3 the sliding and incremental model are capable of dealing with this concept drift. The situation is different after peak #3, indeed in all the subsequent events we observe a decrease in performances. This may happen due to strong dependency on specific words, which are used in some events, but not in all.

We think that with a biggest timeline and new events for testing and retraining models, better results can be produced with the sliding and the incremental models.

6.5 Distribution of opinion polarity over the classes by month using the sliding model

After selecting the sliding model, we classified the tweets of a specific peak with the model trained with the previous peak's tweet and then we grouped the obtained results per month as shown in the picture below. In this way, they can be compared with graphs obtained with the original model.



As we could expect, the model classifies the half tweets per month as 'Neutral', due to elevated presence of journalistic tweet, agencies etc. For the other classes, except for July and August, in which the percentage is quite similar, the model has classified more 'Negative' tweets than 'Positive'.

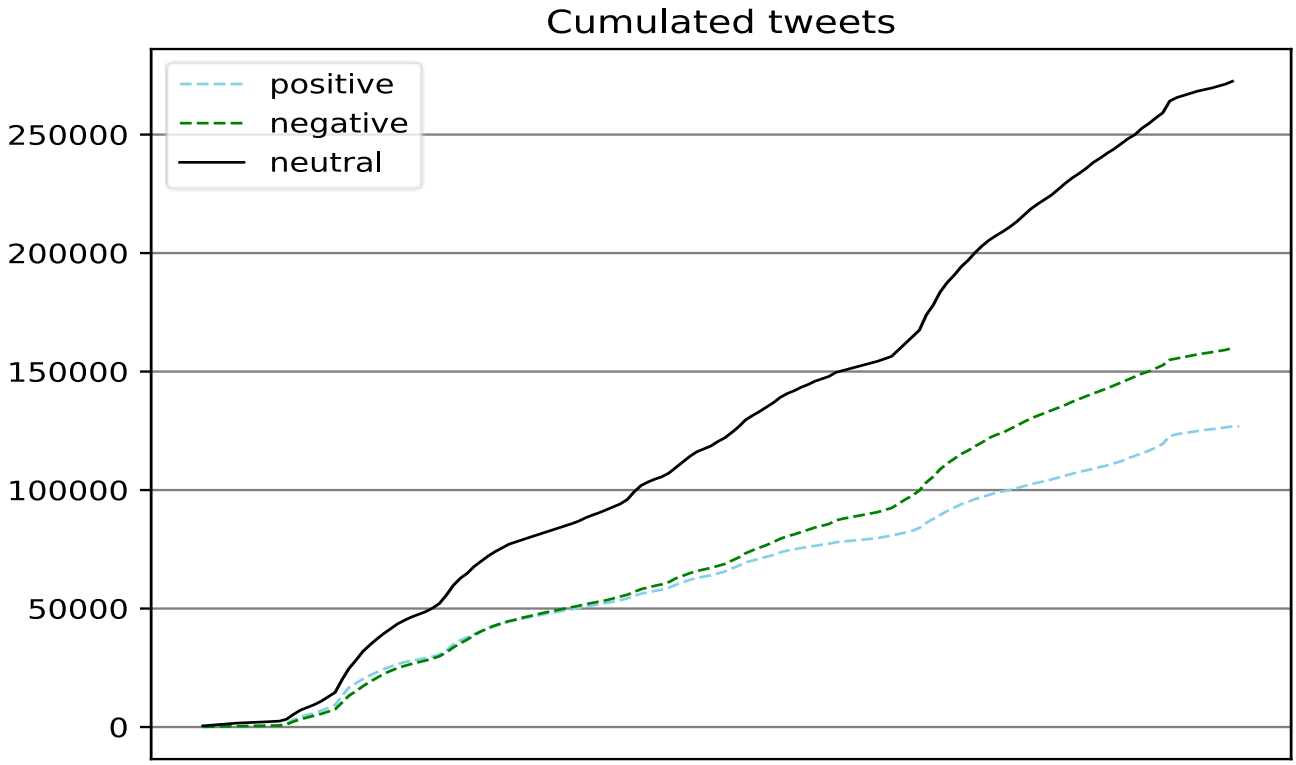
If we compare this figure with the figure in the paragraph 5.1, we can observe how the percentages are different. For example, in September, the original static model classified the 29.5% of tweets analyzed as 'Positive', while with the re-trained sliding model only the 21.77% is classified as positive. The same thing happens in October, but in this case the percentage of 'Negative' tweets is the same, but we observe a larger portion of neutral tweets

6.6 Few examples of tweet classification using the sliding model

In the figure below, we can observe few examples of tweet classification using the sliding model. The tweets come from different peaks. It is possible to notice the case in which the various models guess the right class of tweets – in this case we have a *Hit* -and also cases in which the models do not guess it, maybe due to ironic or journalistic content – in this case we have a *Miss*.

Event	Text of tweet – [English translation]	Actual class	Assigned class	Note
#2	*Manifestazione contro il green pass a Milano: quasi 10mila in corteo – [Demonstration against the green pass in Milan: almost 10 thousand in the procession]*	0	0	Hit
#2	*#GreenpassObbligatorio Sono tra quelli che ancora non ha il green pass (ovviamente per motivi di salute). Nonostante ciò, sono favorevole a questo nuovo obbligo. – [#GreenpassObbligatorio I am among those who still do not have the green pass (obviously for health reasons). Despite this, I am in favor of this new obligation.]*	1	1	Hit
#2	*Scatta l'obbligo del green pass: gestori di bar, ristoranti, palestre e piscine in difficoltà – [The green pass is mandatory: managers of bars, restaurants, gyms and swimming pools in difficulty]*	0	-1	Miss, news tweet
#2	*Come per le MULTE da "violazione dei lockdown" annullate dai Giudici di Pace in tutta Italia, così sarà per il Greenpass incostituzionale. Denunciate tutti coloro che vi richiedono il Greenpass all'ingresso del locale. Peraltro intravedo risvolti penali per violazione privacy. – [As for the "violation of lockdown" FINES canceled by the Justices of the Peace throughout Italy, so it will be for the unconstitutional Greenpass. Report all those who ask for the Greenpass at the entrance of the club. Moreover, I see criminal implications for violation of privacy.]*	-1	-1	Hit
#2	*Se mi costringono al greenpass per motivi di lavoro invece del vaccino mi faccio venire il Covid e presento il certificato di guarigione. Preferisco... dico seriamente. – [If they force me to the greenpass for work reasons instead of the vaccine I get Covid and present the certificate of recovery. I prefer ... I mean seriously.]*	-1	0	Miss
#2	*Il green pass è cmq incostituzionale.....ma i furbacchioni in malafede nn parlano di obbligo x ovvi motivi, ma molti questo non lo sanno,SE ACCETTIAMO OGGI QUESTO,DOMANI FARANNO DI PEGGIO!!!!! – [The green pass is unconstitutional anyway ..... but the crafty ones in bad faith do not speak of obligation x obvious reasons, but many do not know this, IF WE ACCEPT THIS TODAY, TOMORROW THEY WILL MAKE WORSE !!!!!]*	-1	-1	Hit
#3	*Io credo che tutti coloro che si piegheranno a questa norma incostituzionale del greenpass per entrare nei loro esercizi commerciali debbano fallire prima di subito e, personalmente, farò tutto quello in mio potere per boicottare e far boicottare queste attività lavorative. – [I believe that all those who will bow to this unconstitutional greenpass rule to enter their businesses must fail first and, personally, I will do everything in my power to boycott and boycott these businesses.]*	-1	-1	Hit
#3	*c'è sicuramente una categoria fortemente beneficiata dal passaporto vaccinale o #greenpass: gli avvocati che faranno affari d'oro con i ricorsi – [there is certainly a category heavily benefited by the vaccine passport or #greenpass: the lawyers who will do gold deals with appeals]*	0	-1	Miss
#3	*Il #greenpass per entrare al bar e nei ristoranti a me pare un'enorme cazzata inapplicabile !! #DPCM #disobbedisco – [The #greenpass to enter bars and restaurants seems to me a huge inapplicable bullshit !! #DPCM #disobey]*	-1	-1	Hit
#4	*Green pass, Pregliasco: "Bene, ma non escluderei obbligo vaccinale" – [Green pass, Pregliasco: "Good, but I wouldn't rule out the vaccination obligation"]*	0	-1	Miss, news tweet
#4	*I nostri diritti finiscono dove inizia il rischio della vita degli altri. Favorevole al Green pass ✅ #greenpass #vacciniamoci @VoltItalia – [Our rights end where the risk of the life of others begins. In favor of the Green pass ✅ #greenpass #vacciniamoci @VoltItalia]*	1	1	Hit

7 Conclusion



In the picture above, it is possible to observe the cumulated value of tweets by class over the same interval. It can be noticed that this curve increases drastically in correspondence of the various peaks that we have analyzed previously.

In this project, we carried out an analysis regarding people's opinion about the introduction of Greenpass in Italy as consequence of the Covid-19 pandemic through a stance classification of tweets scraped from Twitter.

After testing the various classifiers, we employed the ComplementNB model to classify tweets as belonging to three different classes: in favor, contrary and neutral.

In the end, we reached an accuracy of 66,11%. The results obtained are quite good, but they could be increased by taking in consideration a larger timespan that contains more events.

8 Appendix

8.1 Compare Random Forest with Gradient Boosting:

Classifier	**Accuracy (%)**	Precision (%) by class			Recall(%) by class			F-score(%) by class		
		Pos	Neg	Neu	Pos	Neg	Neu	Pos	Neg	Neu
Random Forest	61,81	59,54	68,12	59,02	71,26	58,75	55,14	64,87	63,09	57,01
Gradient Boosting	61,15	61,38	64,48	57,76	66,43	60,12	56,75	63,81	62,23	57,25

Both the algorithms perform quite well in positive and negative classes, they are slightly worse in neutral tweets prediction. Moreover, we can say that the algorithms perform quite well on the dataset, in fact the results are quite similar, even though Random Forest is slightly better bot to recognize positives and negatives tweets.

Random Forest and Gradient Boosting have not been compared due to their lower performance respect the other classifier.

8.2 Link to the project code

The code of the project is available in a GitHub repository accessible by the following link:

https://github.com/peppoct/SentimentAnalysisGreenpass