FACULTAD DE ESTUDIOS ESTADÍSTICOS

MÁSTER EN MINERÍA DE DATOS E INTELIGENCIA DE NEGOCIOS

Curso 2020/2021

Trabajo de Fin de Máster

TITULO:
Twitter user multiclass classification during US 2020 electoral campaign

***Alumno*: Erol Mrzic**

***Tutor*: Gómez González, Rafael Caballero Roldan, José Manuel Robles Morales**

Septiembre de 2021

UNIVERSIDAD COMPLUTENSE
MADRID

Contents

Abstract

Due to the unprecedented rise of data content on social media over the last decade, an opportunity for data-based analysis has become a norm in the modern world. Implementing Machine Learning algorithms and Data Science methods virtually every industry changed. One of the most active researching areas in Machine Learning today is Natural Language Processing (NLP), a field of Artificial Intelligence (AI) that allows computers to read, understand, and deduce meaning from human languages. In this paper we applied Natural Language Processing methods and algorithms on two Twitter datasets collected during the US 2020 elections in order to group both users and tweets in multiple categories based on their support for the candidate. The purpose of this work was to establish the possibility to correctly classify these individuals and their individual tweets based on their aggregated opinions and to create a predictive classification model focusing on text analysis. As a result, we constructed, trained and tested multiple models that can help predict the probability of the user's sentiment toward the candidates based on their tweets. We showed that in 63 % of the cases, we can present high probability of a user's sentiment classification, according to the amalgamation of their tweets.

Keywords: Data Science; Machine Learning; Sentiment analysis; Multiclass prediction; Natural Language Processing;

1. Introduction

1.1 Motivation

The past decade gave the rise of social networks and online media platforms resulting in towering amounts of online user generated data and a new level of awareness [1], making this the first time in history that we have aggregated amounts of opinionated data recorded in digital form for analysis and research.

This amount of data available encouraged the researchers to develop extensive and more complex models as well as new and innovative use cases [2], [3] in data exploration and data-based predictive models.

Such cases of Machine Learning algorithms together with Data Mining and analysis methods have initiated changes across political campaigns, referendums and governments [2], [4] and their processes. An example was the 2016 US elections [5], between the Democratic candidate Hilary Clinton and the president-elect Donald Trump, where he usage of social media (Facebook) data was used for a "*detailed psychological profiles of every American voter, so that campaigns could tailor their pitches from person to person.*" [6], thus allowing them a clear overview of the voters and helping them create a winning campaign.

Social media platforms, such as Twitter, fundamentally changed not only the way the news is reported and commented, but became an important political communication tool [10]. There has been extensive Twitter analysis research based on user's location and tweet Sentiment Analysis [3], [11] some of which will be included in this work as well. We will take on a fairly new approach by assigning groups based on the type of support that the users show for the candidates.

One of the focuses for this thesis is trying to understand the public mood and opinion towards the US 2020 election candidates, Republican current president Donald Trump and Democratic challenger Joe Biden, using the methods of Natural Language Processing and supervised Machine Learning algorithms.

As the US 2020 election gave rise in economic politic uncertainty [12] and huge polarization between the voters, it presented a great opportunity to analyse the user opinions on social media.

Given that Twitter is the most popular textual content social network today, and as well a favourite way for the now-ex president to communicate with the worldwide population, we have based our research on that specific social network. In order to analyse Twitter data, we will use techniques, models and methods such as Data Mining, Machine Learning and Natural Language Processing.

Data Mining can be best described as the process of discovering patterns and trends in data [7] and can be seen as a combination of several scientific disciplines that provide structure, analytical insights and assistance in making data-based decisions.

Machine Learning is one of the disciplines used during data analysis and involves a set of different methods such as regression and classification to predict future behaviour based on the previous state of data. It has been defined as a science as well as an art or a skill of making computers learn from data [8].

One of the most active subfields in Machine Learning today is Natural Language Processing (NLP) which can be formally described as a component of Artificial Intelligence (AI) that enables a computer program to process input, spoken and written human language, and by converting it to code, successfully read, understand and extract meaning from it in a way that a computer can understand [9].

In this work, we will explain the complete workflow necessary for a Data Mining analysis, which includes data processing and modification, Natural Language Processing methods, visualization techniques and application of predictive Machine Learning models with the aim of getting the probabilities for our target variables. Also, we will see how the predictive models can be improved and refined to be as efficient as possible and provide a higher level of prediction.

1.2 Natural Language Processing

Natural Language Processing (NLP) is a field of Artificial Intelligence focused on making human language understandable to machines. It does so by combining the power of language and computer science to develop systems that can understand and extract meaning from text analysing different aspects such as semantics, syntax and morphology [13].

Using text vectorization, a Natural Language Processing tool, the data will be transformed to a format that the machines can understand and algorithms can use. Text needs to be processed and prepared before the vectorization can take place and this is done by using numerous techniques available such as tokenizing, removal of Stop Words, stemming and lemmatization etc. Our process will be detailed further on in the work where a pipeline was created in order to parse the textual content of the tweets.

Our analysis will be focused on applying Natural Language Processing methods to text patterns and create classification algorithms which will successfully group the users based on their sentiment instead of just classifying the content that they are making.

1.3 Structure

In chapter 2 we will go through the objectives of this work and establish the goals that we want to achieve and which we will measure later.

Then, in chapter number 3 - Methodology, we will explain how the analysis process was designed and executed, detailing every step of the process.

In chapters 4 and 5, Result interpretation and Error analysis, we will look closely at the predictions achieved with the best model and the errors it produced in order to understand how the model works and what are its upsides, flaws and possible betterments.

Lastly, in chapter 6, we will have a general overview of the work and further discussions on future use cases and applications of this analysis.

2. Objectives

In this section, we will introduce our objectives and the give a quick overview of the results which we want to achieve and the necessary steps for it to be done.

2.1 Objective #1

Our data consists of tweets specifically concerning the US elections, which gives us an opportunity to gather interesting insights into the sentiments of the users on this social network.
Therefore, our objective is to extract meaningful information about the opinions for each candidate contained in the tweets from the tweets text content that contains this information. This can be done by filtering the data, in regards to the specific sentiments and for each candidate, and using visualization techniques that will allow us to understand the patterns and trends from the tweets.

As tweets are not solely a textual element, we have to take into consideration mentions (‘@’), hashtags (‘#’) and media links as separate elements in order to get meaningful deductions from them. Analysing these different elements will give us an overview of the supporters for each of the candidates and their focal points of attention.

2.2 Objective #2

Our second objective is to build a classifier that will determine if the message has a positive or a negative meaning. In principle this would result in 4 categories (Trump positive, Biden positive, Trump negative, Biden negative). However, in our case we have the additional neutral category, which gives rise to 9 categories shown in Table 1. The neutral category means that the user has neither positive nor negative opinion, or no opinion at all about the candidates.

The methodology of class assignation, shown in Table 1, is based on the manual classification which has been done previously by “The Data Science and Soft Computing for Social Analytics and Decision Aid Group at the Universidad Complutense de Madrid”.

Classes Assigned	Tweet sentiment – based on manual classification
Biden neutral, Trump negative	No significant mention of Biden, negative comments for Trump
Trump neutral, Biden negative	No significant mention of Trump, negative comments for Biden
Supporting Biden, Trump negative	Positive comments for Biden, negative comments for Trump
Supporting Biden, Trump neutral	Positive comments for Biden, no significant mention of Trump
Supporting Trump, Biden negative	Positive comments for Trump, negative comments for Biden
Supporting Trump, Biden neutral	Positive comments for Trump, no significant mention of Biden
Neutral	No significant mention of both candidates
Negative towards both	Negative comments for both candidates
Positive towards both	Positive comments for both candidates

Table 1 – Class assignation methodology

Newly formed classes, as shown at the left side of Table 1, will be used as a target variable for our multiclass prediction model where the objective will be to assign the probability of the user belonging to one of these new classes.

The “individual tweets” dataset, as it has all 2.410 unique users, will be used as a training dataset where we will construct and optimize our Machine Learning prediction models and our “user tweets” dataset, with 188 unique users and 1.830 tweets, will be used as a test dataset where we will try and assign for each user the probability of belonging to a specific class of supporters.

In order to properly assign the correct class for each unique user in the “user tweets” dataset, we need to take into consideration all of their tweets so the prediction is based on the combination of sentiment throughout all of their tweets and not on each individual tweet.

2.3 Overview

In summary, we have 2 objectives:

- **Objective number 1** is to use the conclusions derived from the classification as an overview of the supporters for each candidate and the focal points of their attention.

- **Objective number 2** is to show probabilities of users belonging to one of the above groups, one that can be used in further research and analysis [1], [3].

3. Methodology

In this section, we will introduce our dataset and the methodology of the analysis. We will continue to explain each step of the process, variable modification, feature construction, application and optimization of the Machine Learning algorithms and the interpretation of the results with error analysis.

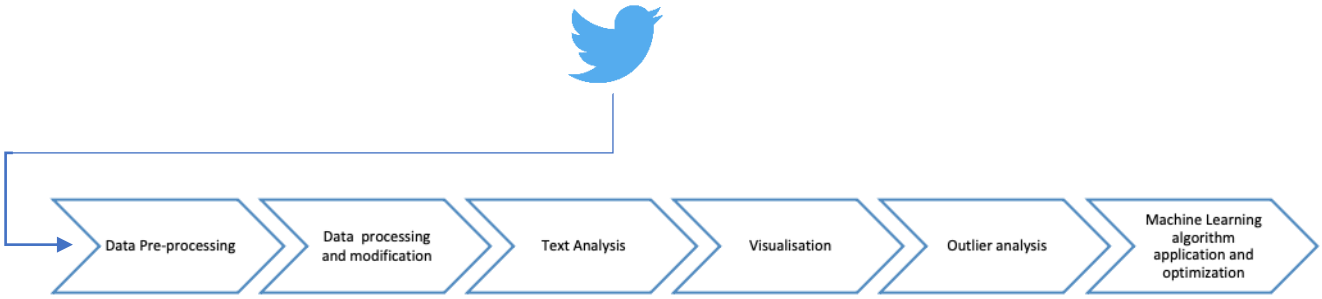


Figure 2 – Methodology process layout

The software used for the analysis is Python, an interpreted high-level general-purpose programming language, currently the favourite choice for data analytics, machine learning, and AI thanks to its vast library ecosystem. The complete analysis has been performed on the Jupyter Notebook environment, which represents an open-source web application where Python code is executed on a local server.

3.1 Data Introduction and Pre-processing

Datasets are often incomplete, inconsistent and are prone to human errors. Data pre-processing is a method of transforming raw data into an understandable, unified and more concise format, this prepares data for further processing using Python programming language.

3.1.1. Data introduction

Our data consists of 2 datasets with 4.754 tweets in total, based solely on the Trump-Biden elections.

With the curtesy of “The Data Science and Soft Computing for Social Analytics and Decision Aid Group at the Universidad Complutense de Madrid”, we have 2.410 manually classified tweets as “Pro Trump”, “Pro Biden” and “Neutral” collected from different users in the period from August to November of 2020, as can be seen on the left side of Table 3.
For future references we will name this dataset “individual tweets”.

The second dataset is an amalgamation of 2.344 tweets from 240 users, stretching from November 2016 to January of 2020, where the same manual classification for candidate sentiment was performed.
For future references we will name this dataset “user tweets”.

3.1.2. Pre-processing

An extensive pre-processing was done to avoid errors during the analysis. The manual classification of the tweet consisted of understanding the textual context and the individuals targeted by the tweet. The media links needed to be checked in case they provided additional sentiment towards the candidates as well as to check if they are still available or not.

Both datasets were then manually revised and filtered to make sure we are only using English language tweets and there are no tweets with illegible special characters which would make the work harder for the algorithm.

For “user tweets” datasets we removed users which had only 1 tweet, as the mean number of tweets per user was 9.8, we took 5 tweets as the minimum number of tweets per user. Total number of tweets removed from the “user tweets” dataset is 514, and the total number of users removed is 52.

Final version of the dataset, which we got after the pre-processing, consists of 1.830 tweets from 188 unique users with the date range from July 2018 up to November 2020, as shown on the right side on the Table 3.

Total number of tweets that we will perform our analysis on is 4.239 with 2.597 unique users from both datasets.

Information for “individual tweets” dataset	Information for “user tweets” dataset
Individual tweets stats Total amount of tweets: 2409 Total amount of unique users: 2409 Earliest date of the tweet: 2020-08-10 23:43:56 Latest date of the tweet: 2020-11-05 08:30:43	User tweets stats Total amount of tweets: 1830 Total amount of unique users: 188 Max amount of tweets per user: 10 Min amount of tweets per user: 5 Average amount of tweets per user: 9.827956989247312 Earliest date of the tweet: 2018-07-31 02:00:39 Latest date of the tweet: 2020-11-05 08:31:59

Table 3 - Statistical information for "individual tweets" and "user tweets" dataset

3.2 Data processing and modification

We import our dataset as .xlsx file in Python where exploratory analysis and modification of the data is performed.

The dataset with individual tweets has columns for ‘id’, ‘created at’, ‘text’ and the manual classification regarding the sentiment of the tweet for Biden ‘B’ and Trump ‘T’ as can be seen on the left side of the Table 4 bellow.

User tweets dataset with the 188 users tweet had the exact same features as the “individual tweets” dataset with the additional ‘user’ column, as it’s shown on the right side of Table 4.

<table>
<tr><th>Individual tweets</th><th>User tweets</th></tr>
<tr><td>(2.410 manually classified tweets collected from different users)</td><td>(1.831 tweets from 188 unique users)</td></tr>
<tr><td>Features:
Unique user ID
Timestamp of the tweet
Tweet content
Manual label for Biden sentiment
Manual label for Trump sentiment</td><td>Features:
Unique user ID
Username
Timestamp of the tweet
Tweet content
Manual label for Biden sentiment
Manual label for Trump sentiment</td></tr>
<tr><td>Example:
<table>
<tr><th></th><th>_id</th><th>created_at</th><th>text</th><th>B</th><th>T</th></tr>
<tr><td>0</td><td>1310942960687689730</td><td>2020-09-29 16:02:12</td><td>@ScottPresler @realDonaldTrump What about inde...</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>1311057930276548615</td><td>2020-09-29 23:39:03</td><td>@realDonaldTrump We are. Voting you out of course</td><td>0</td><td>-1</td></tr>
<tr><td>2</td><td>1311058522105208833</td><td>2020-09-29 23:41:24</td><td>@eril2030 @Cscarb99 @UnitedAsOne2020 @realTyle...</td><td>0</td><td>0</td></tr>
<tr><td>3</td><td>1311058533702565892</td><td>2020-09-29 23:41:27</td><td>@realDonaldTrump @JoeBiden @BarackObama @SenSc...</td><td>0</td><td>-1</td></tr>
<tr><td>4</td><td>1311059375792906240</td><td>2020-09-29 23:44:48</td><td>@toddstarnes @JoeBiden @KXEL1540 Word on the s...</td><td>0</td><td>0</td></tr>
</table></td><td>Example:
<table>
<tr><th></th><th>_id</th><th>usu</th><th>created_at</th><th>text</th><th>B</th><th>T</th></tr>
<tr><td>0</td><td>1324216762972459009</td><td>HamidYoseflI00</td><td>2020-11-05 06:07:33</td><td>Do not leave the president of the hearts alone...\n@realDonaldTrump \n\n#GivebackTrumpvotes https://t.co/nu0nD688xl</td><td>0.0</td><td>1.0</td></tr>
<tr><td>1</td><td>1324068535258226690</td><td>HamidYoseflI00</td><td>2020-11-04 20:18:33</td><td>@realDonaldTrump You must be next president, stay strong 💪\n#GivebackTrumpvotes</td><td>0.0</td><td>1.0</td></tr>
<tr><td>2</td><td>1324101575279890437</td><td>HamidYoseflI00</td><td>2020-11-04 22:29:50</td><td>the bloodthirsty dictator of Iran, who has killed more than 3,000 people in the past year and does not even allow his citizens to use twitter, tweeting to criticize #Trump and the United States. He has gained so much insolence from his Democratic supporters!\n#GivebackTrumpvotes https://t.co/GVplc5nOFU</td><td>-1.0</td><td>1.0</td></tr>
<tr><td>3</td><td>1324173055355023360</td><td>HamidYoseflI00</td><td>2020-11-05 03:13:52</td><td>Hold the line!\n#GivebackTrumpvotes</td><td>0.0</td><td>1.0</td></tr>
<tr><td>4</td><td>1324068816532369409</td><td>HamidYoseflI00</td><td>2020-11-04 20:19:40</td><td>Yeah you are going great Sir! \n\n#GivebackTrumpvotes \n#voterfruad</td><td>0.0</td><td>1.0</td></tr>
</table></td></tr>
</table>

Table 4 –Basic overview of the imported data: “individual tweets” and “user tweets” dataset

In order to properly classify the users with our predictive model, we opted to merge all tweets for each of the unique users in the “user tweets” dataset into one conjoint text. This will enable us to classify the user based on the complete sentiment output of his tweets even thought this can cause problems, such as where the textual context of the tweet and the sentiments toward the candidates get mixed.

It has been a discussion on how to measure the sentiment of users where out of every 10 tweets, there has only been 1 mention for the opposite candidate and it was negative. It was debated that, in fact, we should consider that the user spoke 100% negatively about the candidate even if they only mentioned it once in 10 tweets. This

presents a topic for possible further discussion and betterment of the analysis. In Table 5 we can see an example illustrating this debate and our solution.

As for our analysis, user was marked with having negative or positive sentiment towards the candidate even if they were mention in only 1 out of 10 tweets. The average grade was taken for 'B' and 'T' manual classification and was subsequently converted to either "1" or "-1".

In Table 5 bellow on the left-hand side, we have the definition and an example of calculating the average sentiment for a user based on the manual classification of each individual tweet. We defined average sentiment for users as a sum of all manually classified grades (-1,0-1) divided by the number of tweets for that user.

For example, user having 10 tweets, out of which 3 have been manually classified with negative sentiment for Biden (-1) and 7 tweets that have been manually classified with positive sentiment for Trump (1), will have the average sentiment -0.3 for Biden and 0.7 for Trump.

On the right-hand side, we have the definition and example of calculating the general sentiment for the user based on the average sentiment previously calculated. General sentiment is defined as a rounded average sentiment, rounded up if the average sentiment is positive and rounded down if the average sentiment is negative.

Using the same example as before, where the user has a -0.3 average sentiment for Biden and 0.7 for Trump, the general sentiment for this user, applying our solution, will be -1 for Biden and 1 for Trump.

	Average sentiment for users	**General sentiment for users**
Definition	Sum of all manually classified sentiments for each user divided by the number of tweets for that user	Using Python functions "floor" and "ceil", the average sentiment is rounded up if positive and rounded down if negative.
Example	**User (10 tweets):** • 3 tweets classified as negative Biden (-1) • 7 tweets classified as positive Trump (1) The average sentiment for that user will be: -0.3 for Biden 0.7 for Trump	**User (10 tweets):** • 3 tweets classified as negative Biden (-1) • 7 tweets classified as positive Trump (1) The general sentiment for this user will be: -1 for Biden 1 for Trump

Table 5 – Calculation of general sentiment for merged tweets in "user tweets" dataset

The classes for the users were formed, showing the sentiment of the tweet that corresponds to the combination of the values from manually classified columns “B” and “T” shown on Table 6.

Classes Assigned	B	T
Biden neutral, Trump negative	0	-1
Trump neutral, Biden negative	-1	0
Supporting Biden, Trump negative	1	-1
Supporting Biden, Trump neutral	1	0
Supporting Trump, Biden negative	-1	1
Supporting Trump, Biden neutral	0	1
Neutral	0	0
Negative towards both	-1	-1
Positive towards both	1	1

Table 6 – Classes for users based on the combination of the manually classified sentiment towards the candidates

A new column named “group” was added to both datasets in which the tweets were automatically assigned to their corresponding class based on the sentiment shown in the tweets, as can be seen on the Figure 7.

	_id	created_at	text	B	T	group
0	1310942960687689730	2020-09-29 16:02:12	@ScottPresler @realDonaldTrump What about inde...	0	1	supporting Trump, Biden neutral
1	1311057930276548615	2020-09-29 23:39:03	@realDonaldTrump We are. Voting you out of course	0	-1	Biden neutral, Trump negative
2	1311058522105208833	2020-09-29 23:41:24	@eril2030 @Cscarb99 @UnitedAsOne2020 @realTyle...	0	0	neutral
3	1311058533702565892	2020-09-29 23:41:27	@realDonaldTrump @JoeBiden @BarackObama @SenSc...	0	-1	Biden neutral, Trump negative
4	1311059375792906240	2020-09-29 23:44:48	@toddstarnes @JoeBiden @KXEL1540 Word on the s...	0	0	neutral

Figure 7 – Class assignation for tweets depending on the manually classified sentiment

3.3 Text analysis

Analysis and processing of the tweets was made where we dealt with removing punctuation, extraction of hashtags ('#'), mentions ('@') and media links. After conversion to lower case letters, a clean (lemmatized) version of the tweet was added to the dataset. This analysis procedure was applied to both "individual tweets" and "user tweets" datasets.

Stemming and Lemmatization is a Natural Language Processing technique which reduces the words to their root, removing the suffixes and prefixes and leaving only the actual stem of the word [14]. This allows us to reduce the noise in our data, reducing the linguistic forms to the common base of a word.

For example, if we haven't done this, words such as "wrote", "written", "writing" and "write" would count as 3 separate words, although for the purposes of our analysis, they all have the same base meaning and that is "write".

Difference between Stemming and Lemmatization process, as can be seen in Figure 8, is that Stemming can result in a stem word which actually has no meaning where a lemmatized word will still belong to the language and have an actual meaning.

Lemmatization has proven to be slightly more accurate, although slower, than stemming since it considers the context of the word and converts the word to its meaningful base form [15]. For that reason, we opted for one word lemmatization conversions, whereas there are options for using multiple words (n-grams) to generate a common root word.

Example word	Stemming	Lemmatization
Wrote	Wrote	Write
Political	Polit	Political
Damages	Damag	Damage

Figure 8 – Examples of stemming and lemmatization

In Figure 9, we see the results of the process, converting the clean text, seen in column "text", to a stemmed and lemmatized text seen in columns "stemm tweets" and "lemm tweets".

	text	stemm tweets	lemm tweets
0	[scottpresler, realdonaldtrump, what, about, i...	[scottpresl, realdonaldtrump, what, about, ind...	[scottpresler, realdonaldtrump, what, about, i...
1	[realdonaldtrump, we, are, voting, you, out, o...	[realdonaldtrump, we, are, vote, you, out, of,...	[realdonaldtrump, we, are, voting, you, out, o...
2	[eril2030, cscarb99, unitedasone2020, realtyle...	[eril2030, cscarb99, unitedasone2020, realtyle...	[eril2030, cscarb99, unitedasone2020, realtyle...
3	[realdonaldtrump, joebiden, barackobama, sensc...	[realdonaldtrump, joebiden, barackobama, sensc...	[realdonaldtrump, joebiden, barackobama, sensc...
4	[toddstarnes, joebiden, kxel1540, word, on, th...	[toddstarn, joebiden, kxel1540, word, on, the,...	[toddstarnes, joebiden, kxel1540, word, on, th...
...	...	...	...
2404	[taylorloverme, joebiden, honey, i, didn't, tr...	[taylorloverm, joebiden, honey, i, didn't, tri...	[taylorloverme, joebiden, honey, i, didn't, tr...
2405	[scottpresler, realdonaldtrump, fountainheidi,...	[scottpresl, realdonaldtrump, fountainheidi, b...	[scottpresler, realdonaldtrump, fountainheidi,...
2406	[scottpresler, realdonaldtrump, fountainheidi,...	[scottpresl, realdonaldtrump, fountainheidi, b...	[scottpresler, realdonaldtrump, fountainheidi,...
2407	[bigdanielenergy, ionwirlz, 4eagles, drpaulgos...	[bigdanielenergi, ionwirlz, 4eagl, drpaulgosar...	[bigdanielenergy, ionwirlz, 4eagles, drpaulgos...
2408	[joebiden, i, am, sorry, for, the, damages, an...	[joebiden, i, am, sorri, for, the, damag, and,...	[joebiden, i, am, sorry, for, the, damage, and...

Figure 9 – Demonstration of text purification process and use of Stemming and lemmatization techniques

Removing "Stop Words" is considered a routine step in the sentiment analysis. These are words that can be removed from our textual content without any consequences to the prediction model.

This is done by using Natural Language Processing tools, where a list of the words for all global languages has been made containing these "Stop Words". The tool would be then applied to the text and the "Stop Words" would be excluded in the clean version of the output.

However, in some cases the removal of these "Stop Words" can change the meaning of the text [16], and given that these tweets represent a small textual representation of the user's sentiment, we left these "Stop Words" in the tweets and they are a part of the cleaned version of the tweets.

3.4 Visualisation

We can now explore most common tweets, mentions or any interesting variable or pattern noticed during the analysis. We can use visualisation techniques to identify outliers and any missing or faulty data as well.

3.4.1. Overview for all tweets

Before we move on to individual datasets, we will have a quick overview for all 4.239 tweets and their users to introduce the datasets.

As shown in Figure 10, out of our 9 possible classes, only 7 are represented in our data. The class “negative both” is an outlier and “supporting both” is non-existent. We will deal with these outliers in the future.

We can see that the “neutral” tweets are by far the most common occurrence in our data with 2273 tweets classified as not having any sentiment towards Trump or Biden. The following two most popular classes are exclusive to Trump, with 738 tweets with “negative” sentiment toward Trump and 422 tweets with “positive” sentiment toward Trump with Biden neutral sentiment. The third most popular class is the class with “negative” Biden sentiment and no sentiment shown for Trump.

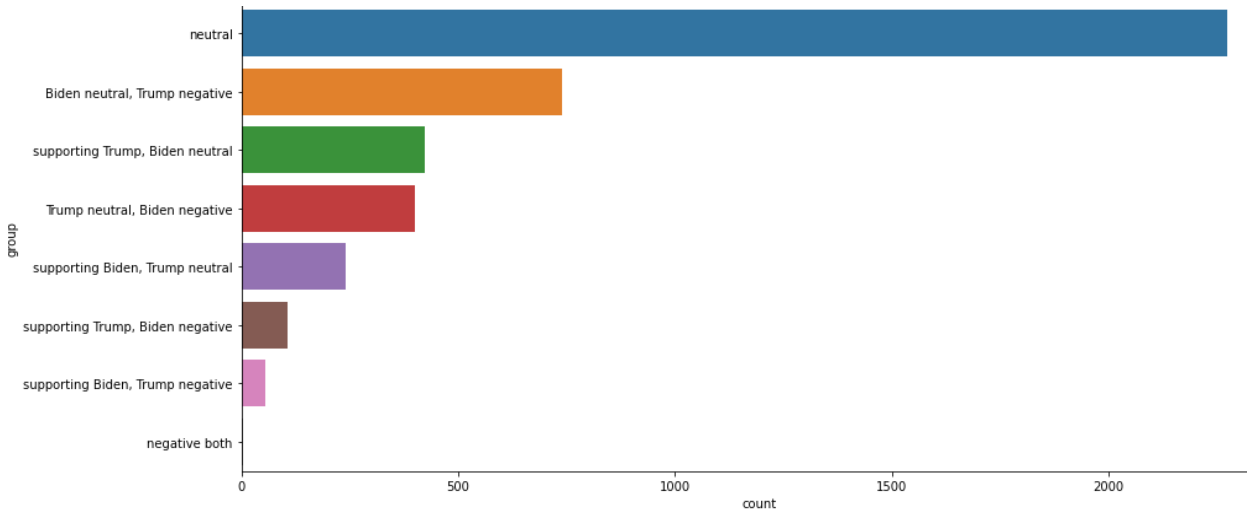


Figure 10 – Bar plot showing class distribution for all tweets

We can already conclude that Trump has a bigger presence on Twitter than Biden, wheatear it be with “negative” or “positive” sentiment.

Overall view of the support towards the candidates, as shown on the Figure 11 and 12 bellow, is distributed equally, with “neutral” tweets as the most common occurrence, making it over 80% of the cases for Biden and almost 70% for Trump.

Another interesting fact is that both candidates have more “negative” than “supportive” tweets. As shown in Figure 10, Biden has 7% positive tweets and 12% negative sentiment tweets with 3.433 Biden tweets marked as “neutral”

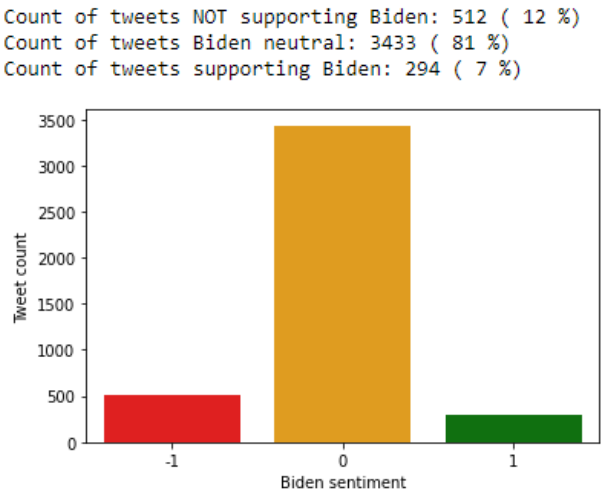


Figure 11 - Count plot showing support distribution for Biden in all tweets

As for Trump, 2.914 tweets were marked as “neutral” with 12% positive and 19% negative sentiment tweets, as shown in Figure 12.

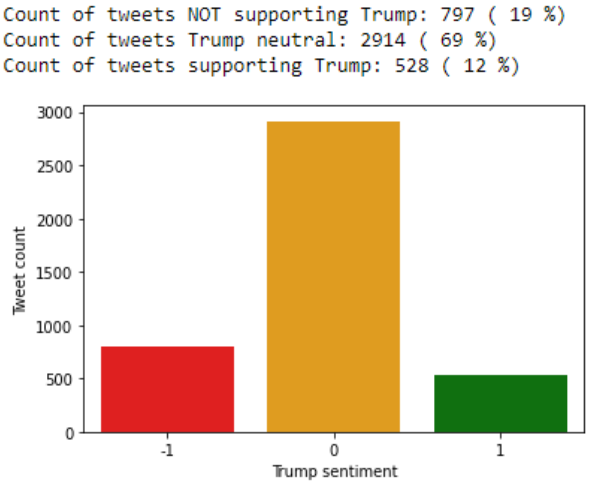


Figure 12 - Count plot showing support distribution for Trump in all tweets

With Trump having 797 tweets marked as “negative” and 528 marked as “positive” and Biden having 512 tweets marked as “negative” and 294 as “positive”, we can conclude that Trump has tweets showing more extreme sentiment than Biden.

The vast majority of our tweets is coming from October and November 2020, as shown on the Figure 13.

We can see a gradual increase in October and the peak on 3rd of November, when the actual election took place and the vote count was in session. At that point, Trump "supportive" tweets were the most popular ones for the first time in our dataset, followed closely by the "negative" Trump tweets. Biden tweets were the least popular tweets at that peak on November 3rd.

We see that Trump had a much larger tweet count both "supportive" and "negative" than Biden. At the highest point Trump had 168 "supportive" tweets and 122 "negative" tweets, where Biden had 96 "negative" and 76 "positive" tweets.

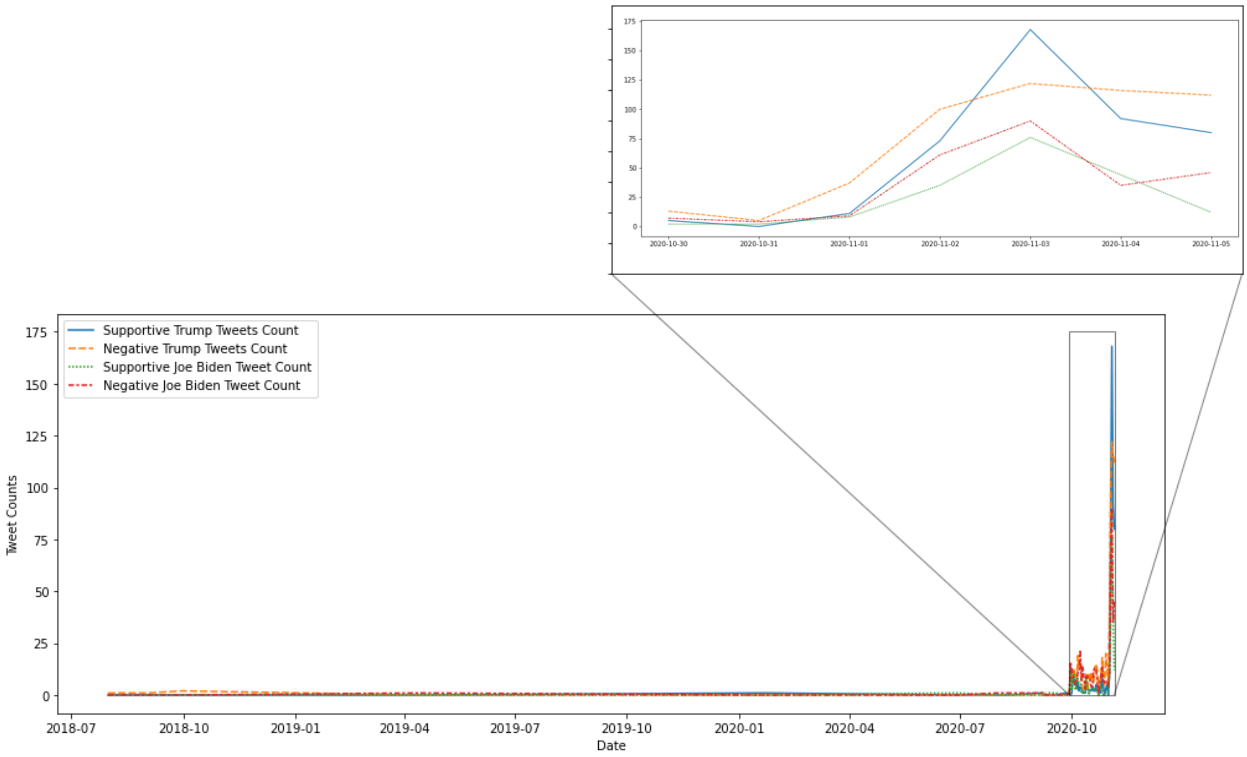


Figure 13 – "Supportive" and "Negative" tweet count over time for both candidates

Although the "negative" tweets regarding Trump remained at almost the same level after the peak, the "negative" tweets regarding Biden fell and then experienced a new rise.

Supportive tweets for both candidates fell after the peak, with Trump tweets still significantly above the Biden "supportive" tweets.

3.4.2. Most Common Words

Analysing the most common words for tweets with “wordcloud” gave us an interesting insight in the topics of the voters, as shown In Figures 14 and 15.

Trump supporters were very focused on the legality of the elections with most popular tweets like “rigged election”, and “givebacktrumpvotes” and Biden negative tweets had a very strong focus on linked individuals such as Kamala Harris, Barack Obama and Hunter Biden.

Most common words in negative tweets - TRUMP	Most common words in positive tweets - TRUMP
	

Figure 14 – Most common used words in tweets with negative and positive sentiment regarding Trump

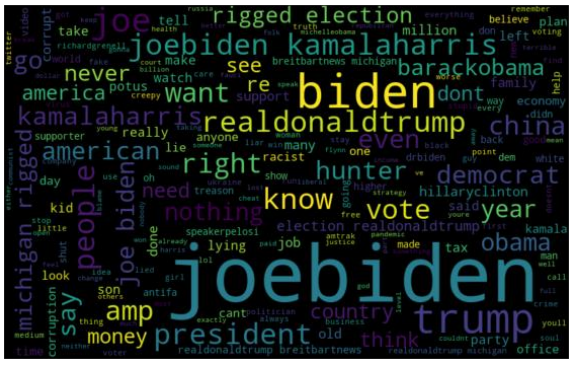
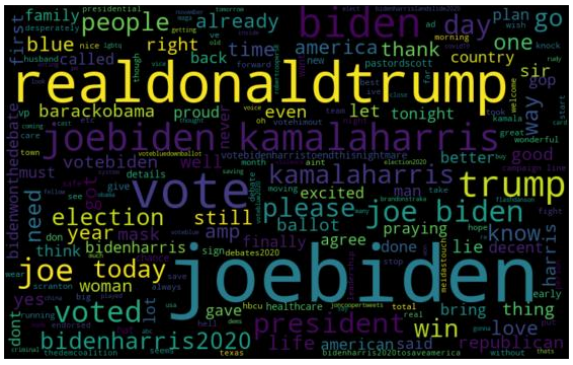
Most common words in negative tweets - BIDEN	Most common words in positive tweets - BIDEN
	

Figure 15 - Most common used words in tweets with negative and positive sentiment regarding Biden

3.4.3. Most Mentions – "@"

Most "mentions – @" in tweets were Biden and Trump for both datasets, which was to be expected, but interestingly Biden supporters were much more inclusive of other individuals than Trump supporters in their tweets.

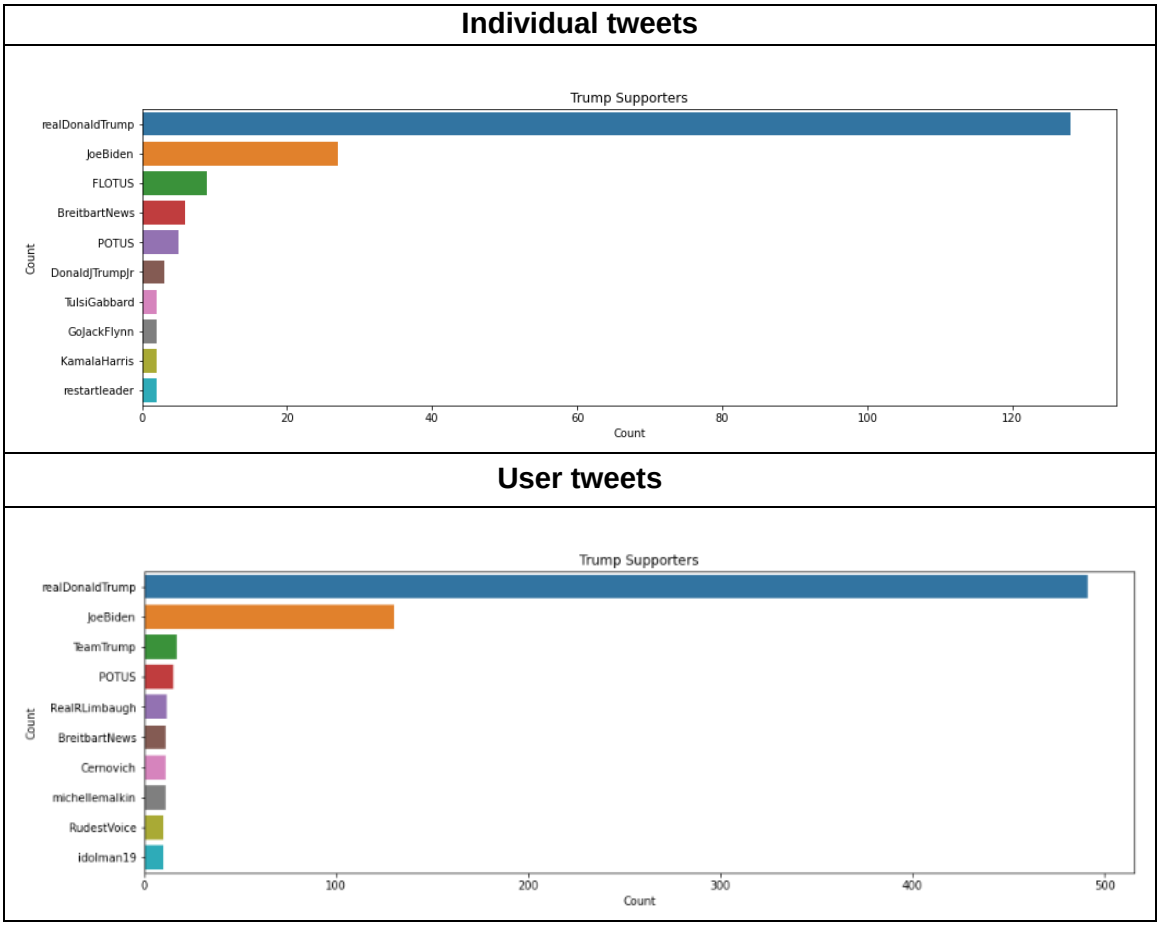


Figure 16 – Most mentions in Trump tweets in "individual tweets" tweets (top) and "user tweets" (bottom) dataset

As seen in Figure 16, Trump supporters feature significantly Donald Trump with Biden as second most popular but with the ratio 5:1. Other prominent mention in both datasets is the Breitbart news, a far-right syndicated news which leaned heavily on the Trump campaign.

We can see some inconsistencies as well, where in the "individual tweets" dataset the term FLOTUS (First Lady Of The United States), signifying the wife of the now-ex president Trump - Melania Trump, was the 3rd most popular mention, does not even show in the "user tweets" dataset.

As for the Biden supporters, as shown in Figure 17, in the “individual tweets” – Joe Biden is the most popular mention ahead of Donald Trump, where in the “user tweets” we see again Donald Trump ahead of Joe Biden.

Both dataset feature often Kamala Harris, which is now the first woman to be elected as vice president in the US history and was the 3rd most popular mention.

Other prominent mentions are ex-president Barack Obama, for which Joe Biden performed the duty as vice-president for two terms during 2008 – 2016 period, Douglas Emhoff, the husband of Kamala Harris and now the first gentleman of US, and Lady Gaga, an artist which was very vocal about the Biden campaign and performed at the inauguration.

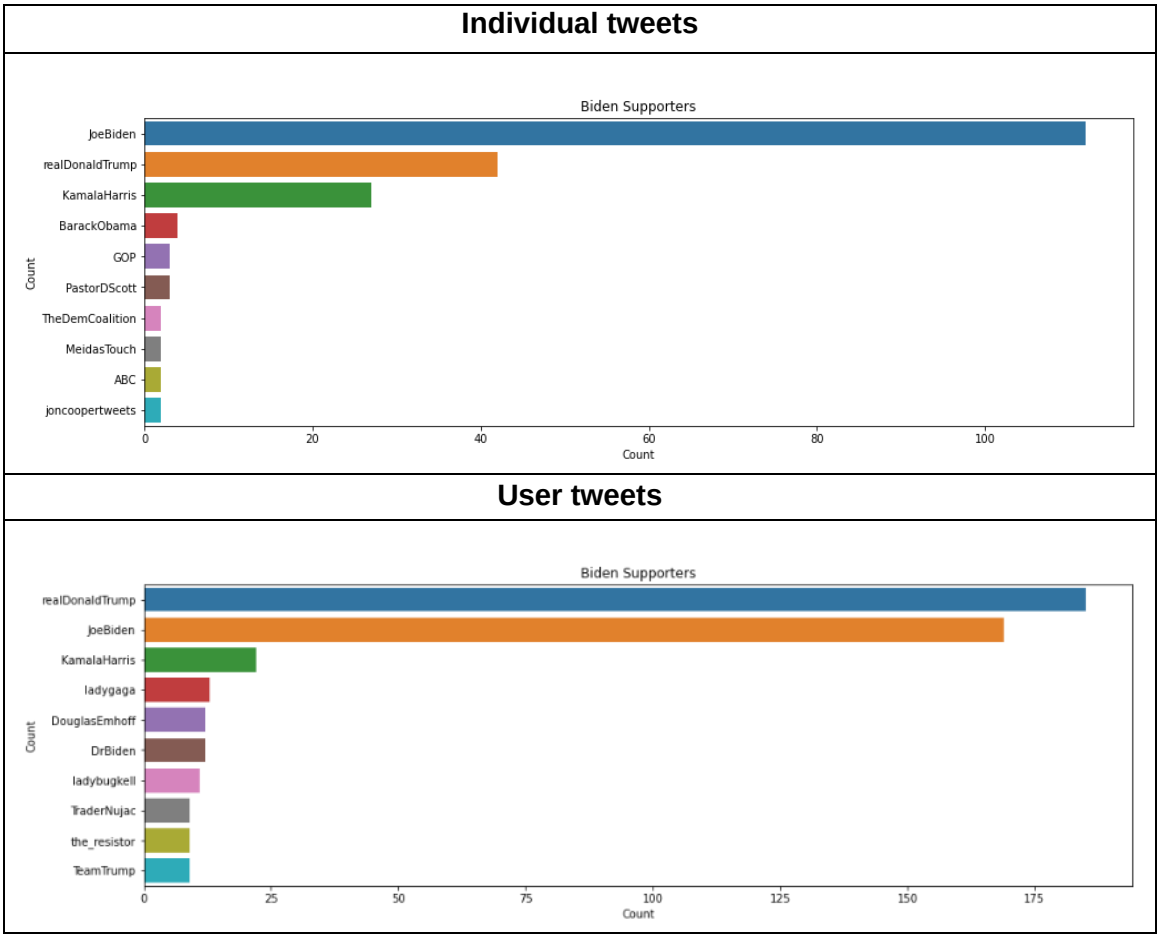


Figure 17 – Most mentions in Biden tweets in “individual tweets” tweets (top) and “user tweets” (bottom) dataset

3.4.4. Media Links

Media links appeared to be too ambiguous and most of the time inaccessible, so they will not be included in the analysis. This does represent a drawback, given that media links are sometimes the main focus of the tweet and are crucial for understanding the sentiment of the tweet. In Figure 18 we can see the distribution of tweets with and without the media links in both datasets.

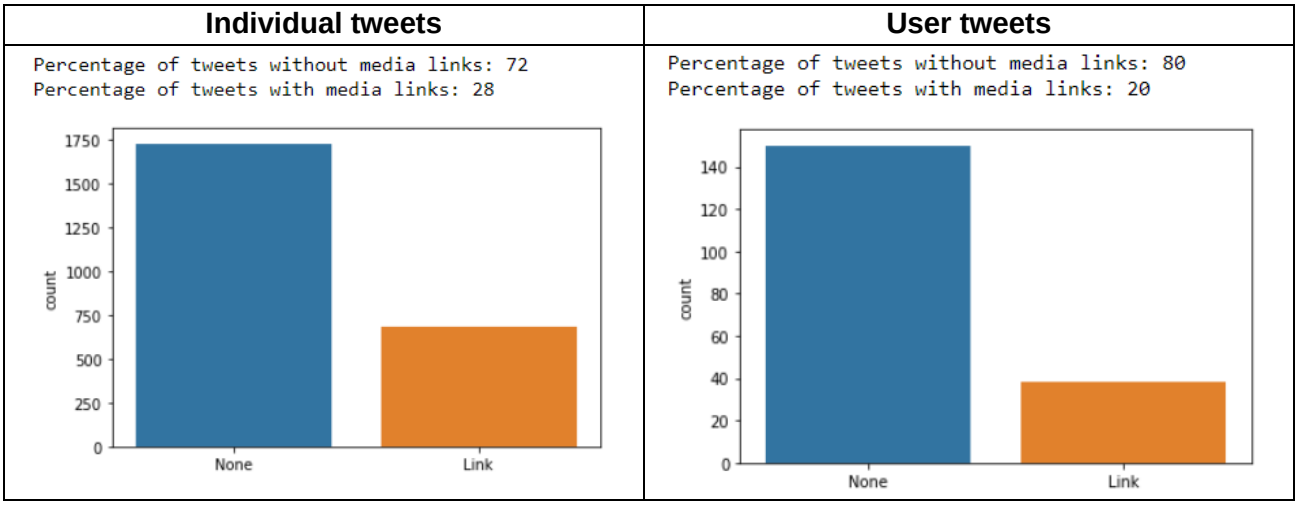


Figure 18 – Number of tweets with Media links in "individual tweets" and "user tweets" dataset & their percentage

3.4.5. Target class distribution – user groups

Visualizing the new made column 'groups' in Figure 19 and 20, where the assigned classes are sorted based on the number of users, we have an important overview of the class distribution and possible outliers for both datasets.

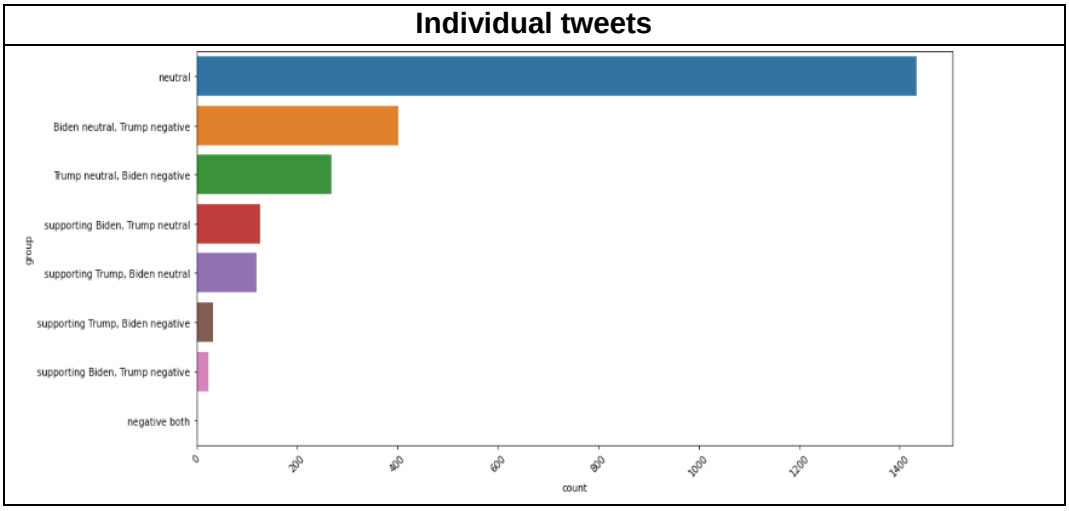


Figure 19 – Bar plot showing class distribution on "individual tweets"

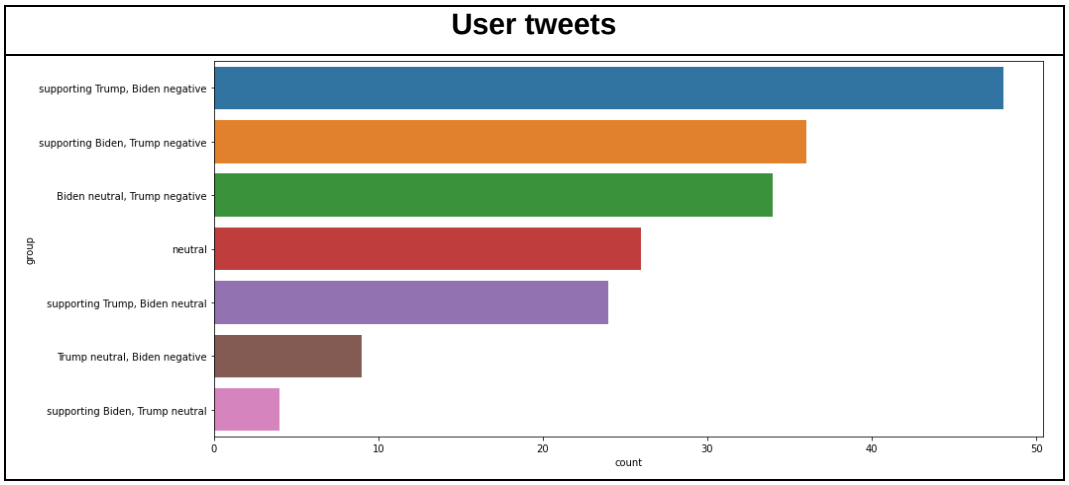


Figure 20 – Bar plot showing class distribution on "user tweets" datasets

We could see in Figure 19 and 20 that the 'negative both' group is an outlier in the "individual tweets" dataset and the 'supportive both' is non-existent in both, so we removed the two groups from both datasets, as they would not be representative enough to make an impact to the analysis.

As this is a competitive election between two candidates, these two outlier groups are to be expected and prove to show that the data is mirroring what we saw in real life, as the majority of users formed one sided opinion on the candidates.

After modifying our user groups, we still have an imbalanced classification problem. An imbalanced classification problem is an example where the distribution of data across our classes is biased or skewed. The distribution might vary from a severe bias to a very slight bias, where we could have a ratio of one datapoint for the minor class against hundreds or thousands of datapoints in the major class or classes.

3.5 Solving imbalance class problem

Imbalanced classes represent a challenge for our predictive models, as the algorithms were designed around the assumption of an equal number of datapoints for each of the classes. This will result in a model that will either ignore the minor classes or have a very bad predictive performance for them.

We will use SMOTE (Synthetic Minority Oversampling Technique) technique and stratify sampling method (proportional division of data) to make a balanced distribution and fill in the minority classes in the training set when fitting our prediction models.

SMOTE technique is a type of data augmentation that synthesizes new samples from the existing ones selecting samples in the minority class and duplicating them. As a result, no new information is brought to the dataset and the distribution between the classes is balanced [17].

Advantages of using this technique than just regular resampling, which would in turn replicate the same instances of minority classes, is that it avoids overfitting of the model, as the new data is not replicated but rather generated to mimic the minority classes. This would in turn, avoid the model to just “learn” the replicated training data and cause overfitting. Advantage over the under-sampling, which would remove the data from the majority classes until the distribution is balanced, is that we don’t lose any useful information.

However, the major disadvantage is that we are generating more noise in our data which can then affect the performance of the model.

3.6 Machine Learning algorithm application and optimization

After the data has been modified and outliers have been removed, we need to prepare the data for Machine Learning model application. This means having the data in correct format so that the algorithms can learn and make predictions based on it.

Once we have our data prepared, we can start applying different algorithms and optimizing the hyperparameters in order to achieve the best performance of the prediction models possible.

3.6.1. Overview

We need to split our data into training and test datasets so that our model can avoid overfitting and can be evaluated appropriately on the test dataset.

The objective of this work was to classify the users with the merged tweets from the “user tweets” dataset, meaning that “user tweets” represents our test dataset where “individual tweets” represent our training dataset.
However, the “user tweets” dataset have their users with tweets merged as to the “individual tweets” dataset where we have only one tweet per user, making the prediction altogether simpler without the problem of losing context of the tweets or mixing sentiments toward the candidates.

We decided that we should perform the test on both of our datasets, both “individual tweets” and “user tweets” so we can have a good overview of the best performing model. We will train and test the models on the “individual tweets” dataset first, splitting it into train and test sets and afterwards they will be tested then on the “user tweets” dataset.
Consistent good performance on both datasets will be the factor on which we will choose the best model.

3.6.2. Data Preparation

A pipeline was created (Figure 21 and 22) where TF-IDF Vectorizer will be fitted transforming our lemmatized word list to feature vectors, SMOTE technique will be applied on the classes and where finally the model would be trained and tested.

TF-IDF (Term Frequency — Inverse Document Frequency) reflects how relevant a word is in each document and is used to transform text into a meaningful representation of numbers which is mandatory to fit Machine Learning algorithms for prediction.

Among other techniques for transforming text to a proper format there is a “bag of words” method, which is a set of vectors representing the count of words in the document not taking into consideration their importance and adding weights to the words, like TF-IDF method. In this work we focused on the usage of TF-IDF as the core transformation methodology for the analysis.

As per our newly formed classes, this represents a multiclass classification problem, so the parameters for each algorithm will be set accordingly.

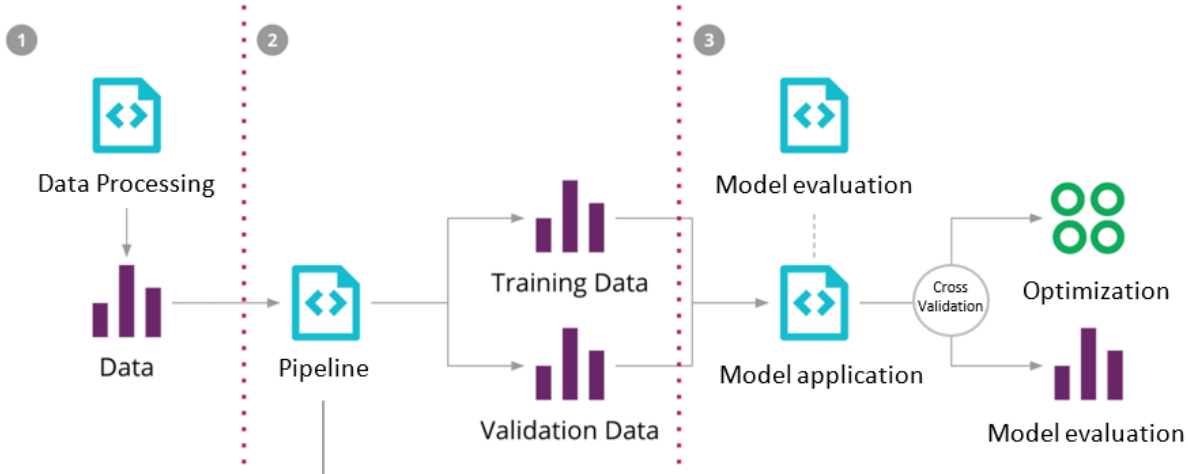


Figure 21 – Model application process methodology

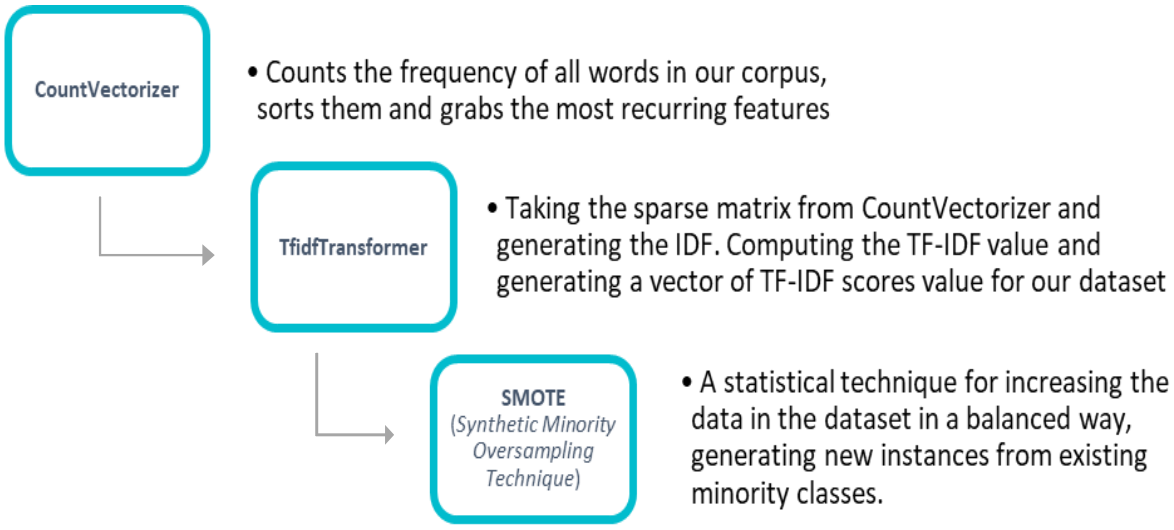


Figure 22 – Pipeline process

3.6.3. Parameter Optimization

For parameter optimization we used Grid search which is a model hyperparameter optimization technique. In scikit-learn this technique is provided in the GridSearchCV class.

When constructing this class, we must provide a dictionary of hyperparameters to evaluate. This is a map of the model parameter name and an array of values to try. In our case we will optimize our score based on “accuracy”.

By setting the “n_jobs” argument in the GridSearchCV constructor to “-1”, the process will use all cores on our machine. The GridSearchCV process will then construct and evaluate one model for each combination of parameters. Cross validation is used to evaluate each individual model and the default of 3-fold cross validation is used.

Once the optimum parameters were found for each model, the f1-score test had been performed on the same train, test split for all. That way we could be sure that the models are compared to the same set of data before going to Cross Validation with all of the optimized models in order to see what is the performance over different splits for train and test.

3.6.4. Model application and results

We will apply the models, shown in Table 23, on our test-train split of the "individual tweets" datasets and find the optimum parameters and then we will use the "individual tweets" as our training dataset and "user tweets" as our test dataset. That way we can compare the results of the models predicting both individual tweets from "individual tweets" and the merged tweets from "user tweets" dataset and take the most constant model as the best.

Model Name	Model description
Logistic Regression	statistical model using Logistic function to model the conditional probability
Random Forest	Ensemble of Decision Tree algorithms which are a flowchart like structure of nodes (features)
Naive bayes	Collection of algorithms based on the Bayes' Theorem
SGD Classifier	Optimization algorithm based on minimizing the loss function with Log Reg and SVM algorithm
Support Vector Machines	Discriminative classifier defined by a separating hyperplane
Gradient Boosting	Decision Tree ensemble algorithm with gradient descent
Bagging	Bootstrap aggregating ensemble of predictive algorithms
XGB	Extreme Gradient Boosting with more accurate approximations in order to find the best model

Table 23 – List of models used in the analysis with brief description for each

The results showing the performance of each model and their parameters can be seen on Table 24 for the "individual tweets" and for "user tweets", where for each of the models we have their optimum parameters found after our GridSearchCV constructor in the column *"Optimum model Parameters"* and their F1 results on the same random state split, thus ensuring that all the models were trained on the same data, and finally the results of the model after a 5-fold Cross Validation.

Name of the ML model	Optimum model Parameters	"Individual tweets" dataset		"User tweets"
		F1- score on same random state split	F1- score on 5-fold Cross Validation	F1- score on 5-fold Cross Validation
Logistic Regression	multi_class = 'multinomial' solver = 'lbfgs' max_iter = 1000	0.54	0.60	0.32
Random Forest	bootstrap = False max_depth = 150 max_features = sqrt min_samples_leaf = 1 min_samples_split = 2 n_estimators = 200	0.60	0.62	0.35
Naïve bayes	alpha = 0.01	0.45	0.58	0.30
SGD Classifier	alpha = 0.0001 max_iter = 100 penalty = none	0.51	0.57	0.30
SVM	C = 200 gamma = 0.01 kernel = rbf	0.50	0.59	0.33
Gradient Boosting	learning_rate = 0.5 max_depth = 5 max_features = 5 n_estimators = 90	0.49	0.49	0.30
Bagging	Base_estimator_C = 100 max_features = 0.9 max_samples = 0.5	0.47	0.61	0.33
XGB	objective="multi:softmax" n_estimators= 104 max_depth= 11 learning_rate= 0.0479 subsample= 0.66624 colsample_bytree= 0.74778 gamma= 0	0.57	0.60	0.37

Table 24 – Model performance on "individual tweets" and "user tweets" dataset after optimization of the parameters

We can see that every model performed slightly better after the cross validation. The accuracy is around 60% (Figure 25) for "individual tweets" and around 30% (Figure 26) for "user tweets" but given that we have 7 classes to correctly classify, this is to be expected. Our goal will be to show probabilities of the users assigned to the classes and not an actual prediction.

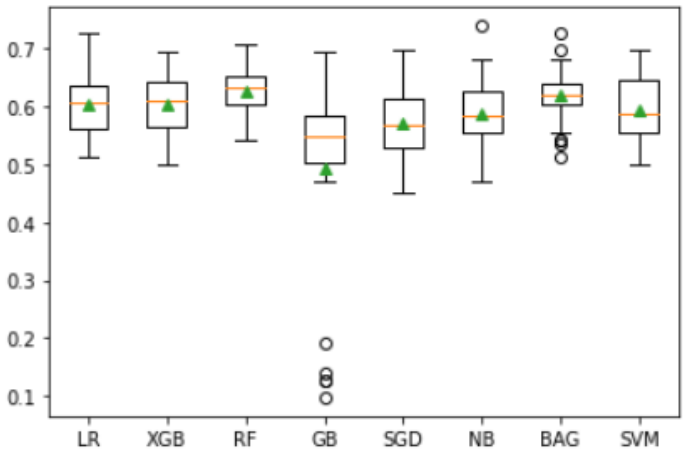


Figure 25 - *Results of the cross validation of the optimized models for "individual tweets" dataset;*

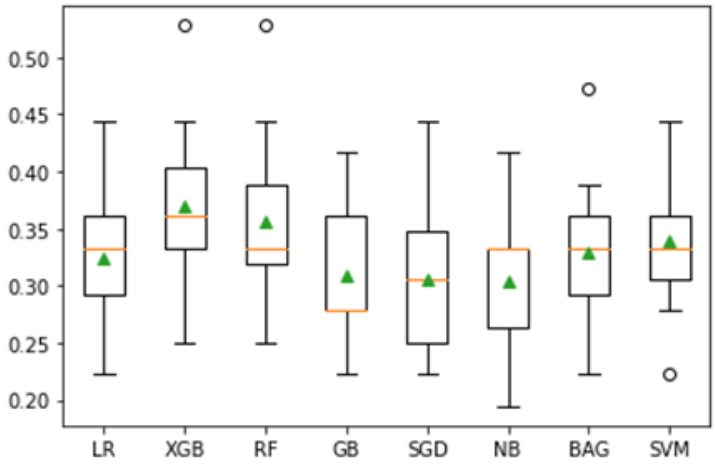


Figure 26 - *Results of the cross validation of the optimized models for "user tweets" dataset;*

Best performing model on the "user tweets" dataset was XGBoosting with 0.37 F1 score, which performed very well on the "individual tweets" dataset as well with a 0.60 F1 score. Random forest model again performed very well, with 0.35 F1 score and like the XGboosting model, has a constant performance with both datasets where it scored 0.62 on the "individual tweets" data.

We can say that XGBoosting and Random Forest were overall best performing models, as they have constant good results, so the final analysis and results were made with the optimum model of the XGB and Random Forest algorithms where we analysed the performance of the models as well as the errors.

4. Result interpretation

As this was a demanding dataset with a multiclass classification problem where the users are meant to be classified based on their opus of tweets, we can expect that the context and sentiment of those merged tweets are sometimes difficult to distinguish and can be mixed for each of the candidates. Thus, the conventional metrics such as accuracy, f1 score and ROC can give misleading impressions over the true efficiency of our models.

So, in order to have a better insight into the performance of both models, we can try and rate their performance based on the probability with which the model assigned classes to the users and the ranking of those probabilities.

Final data frame was constructed with the columns showing the users probability of the correct class assigned and its rank in the prediction as well as the class that the model predicted for the user.

The columns in the new data frame are the username, the correct class that was assigned based on the manual classification results, the model's probabilities of predictions including the class that the model assigned to the user and the probability of the correct class assigned earlier with manual classification. Finally, we have a column that marks the rank of the correct class in the model's predictions. The structure can be seen on Table 28 and 29 together with the individual column explanation in Figure 27.

Column name	Definition
User	Name of the user for which the data belongs to
Correct class	The class which was assigned based on the manual classification
Correct class model probability	The probability assigned by the ML model to the correct class
Predicted model class	The class which was assigned by the ML model
Predicted model class probability	The probability assigned by the ML model to the predicted class
Correct class probability rank	The rank of the correct class (sorted probabilities) in the predictions of the ML model

Table 27 – New data frame structure; Individual columns explanation

	user	Correct class	Correct classmodel probability	Predicted model class	Predicted model class probability	Correct class probability rank
0	1776Katherine	supporting Trump, Biden negative	0.0714	supporting Trump, Biden neutral	0.425243	5
1	1rottnbrawd	neutral	0.1576	Biden neutral, Trump negative	0.256407	4
2	AjeebInsaan15	supporting Trump, Biden negative	0.1298	Biden neutral, Trump negative	0.256407	5
3	Alyauger96	supporting Trump, Biden negative	0.0715	supporting Trump, Biden neutral	0.426178	5
4	Anaxsuescun	Biden neutral, Trump negative	0.2839	Biden neutral, Trump negative	0.283868	1

Figure 28 – Final dataset with predictions, probability and rank made by the XGB model

	user	Correct class	Correct classmodel probability	Predicted model class	Predicted model class probability	Correct class probability rank
0	1776Katherine	supporting Trump, Biden negative	0.1050	Biden neutral, Trump negative	0.280000	5
1	1rottnbrawd	neutral	0.1629	supporting Trump, Biden negative	0.356564	3
2	AjeebInsaan15	supporting Trump, Biden negative	0.0950	Biden neutral, Trump negative	0.310000	6
3	Alyauger96	supporting Trump, Biden negative	0.0700	Biden neutral, Trump negative	0.350000	6
4	Anaxsuescun	Biden neutral, Trump negative	0.1850	neutral	0.470000	3

Figure 29 – Final dataset with predictions, probability and rank made by the Random Forest model

Given that with the merged tweets, the texts lose some of their context and therefore have a very fine margin when predicting the actual class assigned, the actual success rate of the model could be determined by looking at how many correct classes were in the top 3 predictions of the model.

This was done by visualizing the distribution of the probability for the classes that were correctly assigned (Figure 30 and 31) and the distribution of the rank for the correct classes (Figure 32 and 33), where the classes with the rank 1, 2 and 3 represented the instances where the correct class was in the top 3 probabilities predicted by the model.

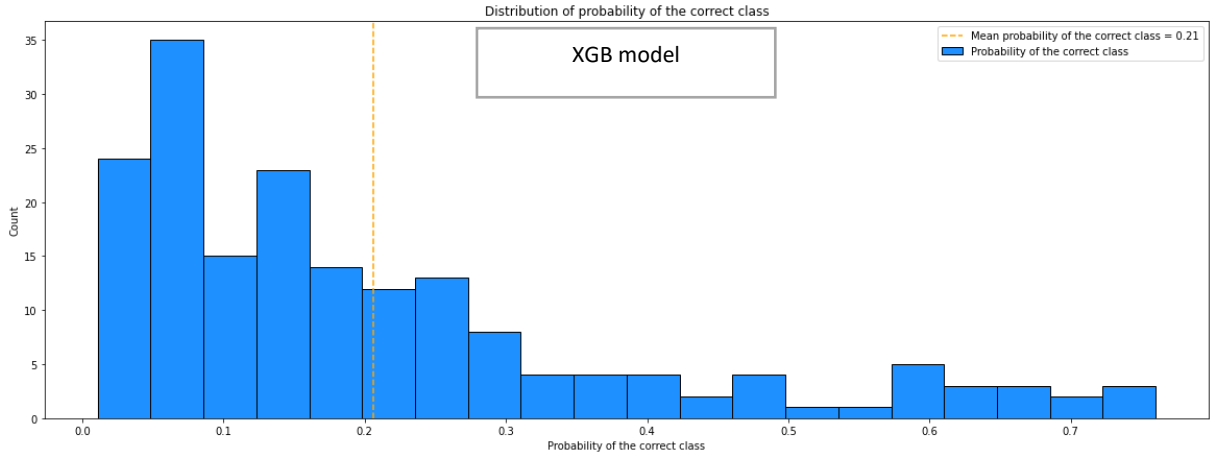


Figure 30 – Distribution of Probability of the class being correctly assigned by XGB model

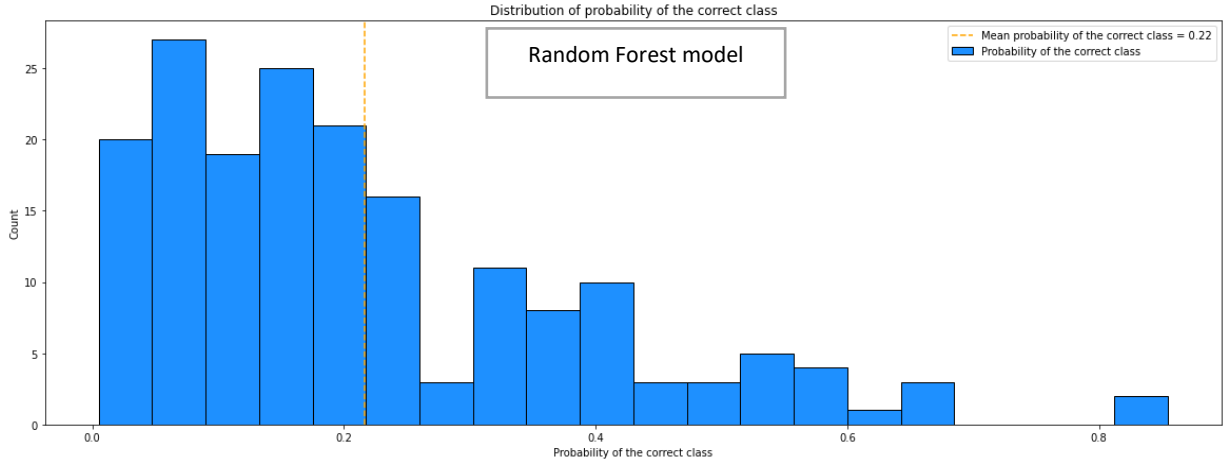


Figure 31 – Distribution of Probability of the class being correctly assigned by Random Forest model

As we can see, both models are performing rather similar with Random Forest model slightly outperforming XGBoosting model on the average prediction probability with 0.22 over the 0.21 for the correct class.

As shown in the Figure 32 and 33, majority of the predictions from both models are ranked in the top 3 predictions, meaning that both models are doing a rather good job of assigning the probabilities for the classes. Slight advantage can be seen for XGBoosting model, as it has the correct class ranked 6th or 7th very small number of times, where Random Forest has the correct class ranked much more as the last option.

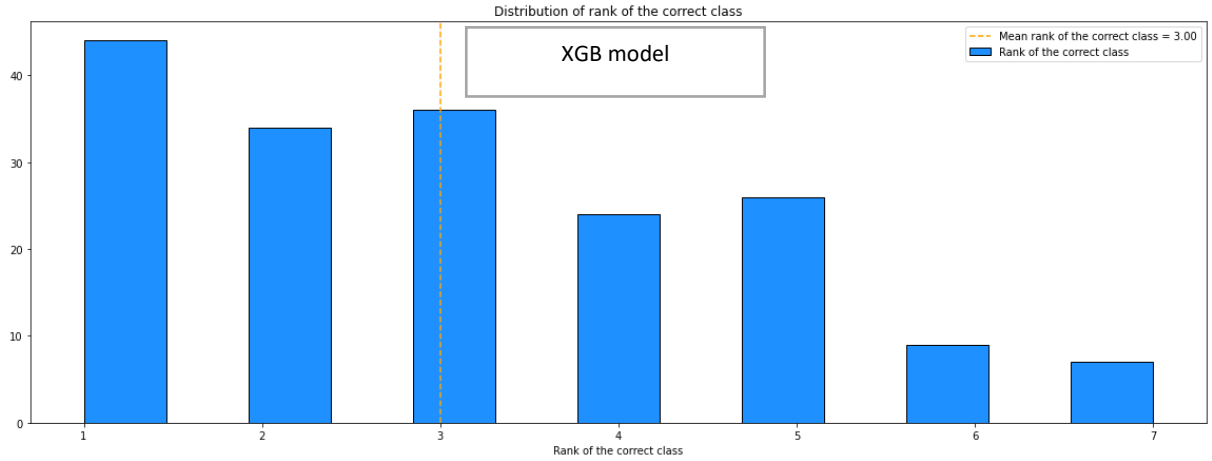


Figure 32 – Overview of the probabilities of the correct class predicted by the XGB model (1 – model predicted the correct class with highest probability, 7 – model predicted the correct class with the lowest probability)

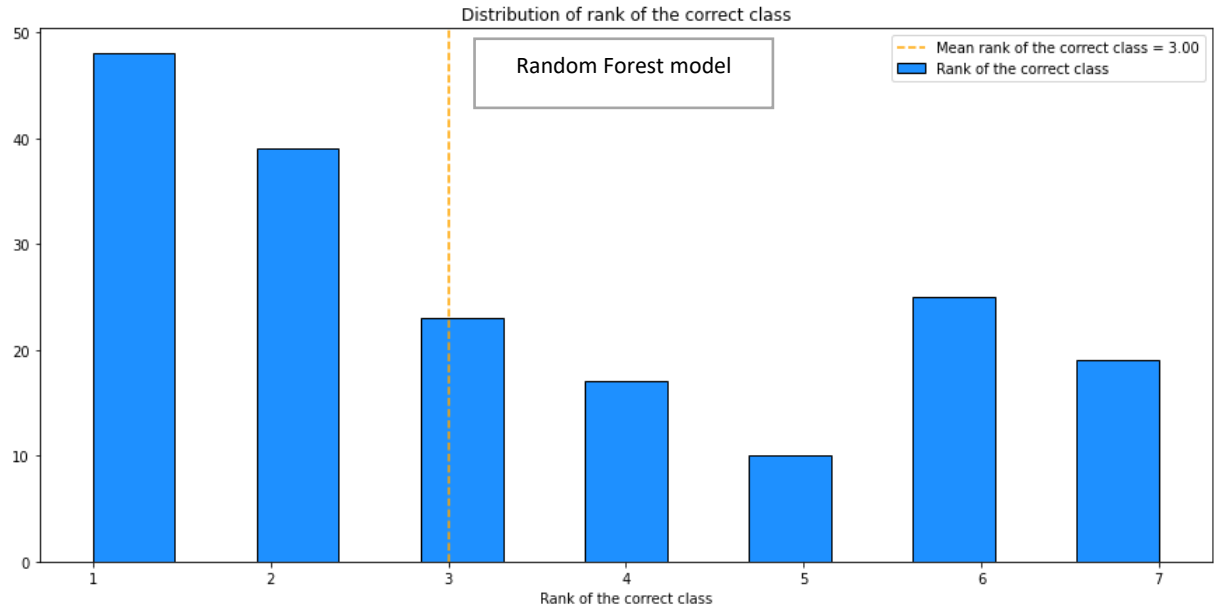


Figure 33 – Overview of the probabilities of the correct class predicted by the Random Forest model (1 – model predicted the correct class with highest probability, 7 – model predicted the correct class with the lowest probability)

Therefore, Random Forest is inclined to misassign the class for the user very drastically, making it a less desirable model for prediction.

The XGBoosting model had the correct class in the top 3 highest probability class prediction in 63 % of the cases, or 114 out of 180 users, as can be seen on the Figure 34. Slightly higher than the Random Forest model where we had 61% top 3 correct class predictions with 110 out of 180 users, as seen in Figure 35.

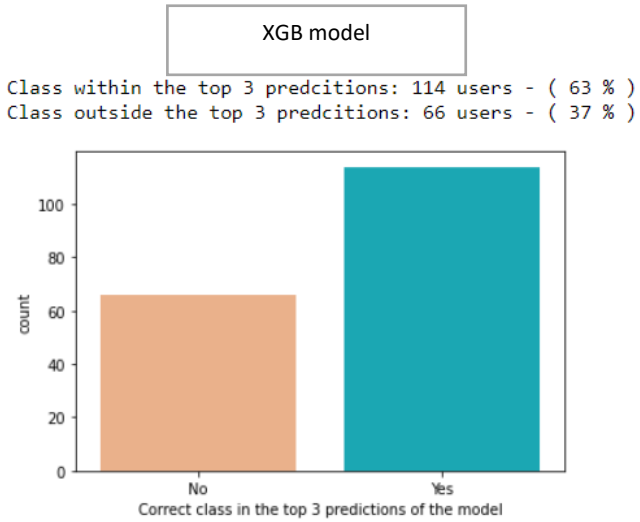


Figure 34 – Percentage of the correct class in the top 3 highest probability predictions of the XGB model

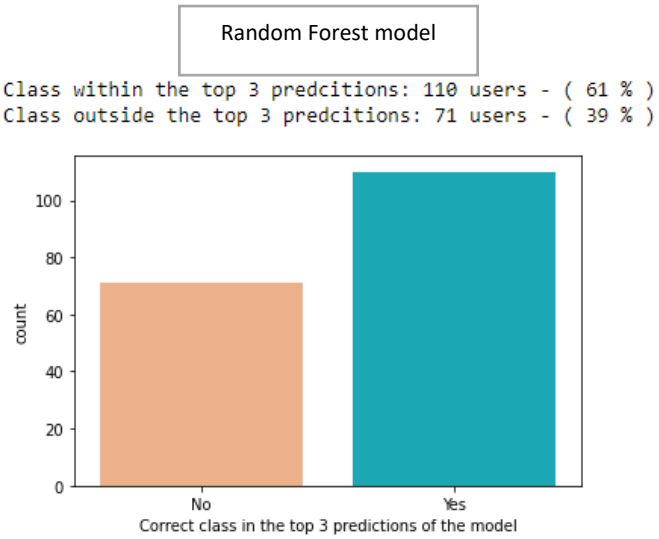


Figure 35 – Percentage of the correct class in the top 3 highest probability predictions of the Random Forest model

In conclusion we can establish that given the higher percentage of correct class probability and the lesser misassign the correct class, the XGBoosting model performed better than any other model we tested.

5. Error Analysis

Given the difference with distribution between the “individual tweets” and “user tweets” classes, as well as the fact that the tweets are now merged for users, it was to be expected that the classification will suffer. The merged tweets present a difficulty, as the data now is mixed, and the sentiment for one candidate might easily be mistaken for another.

Taking a closer look at some of the cases (Figure 36, 37 and 38) where the model was not performing as expected we can get a better insight into the possible reasons of the underperformance.

User

user	Correct class	Correct classmodel probability	Predicted model class	Predicted model class probability	Correct class probability rank	Correct class in the top 3 predictions of the model
1776Katherine	supporting Trump, Biden negative	0.0714	supporting Trump, Biden neutral	0.425243	5	No

User tweets

usu	created_at	text	B	T
1776Katherine	03/11/2020 20:58	@realDonaldTrump Including GLOBALISTS!!!	-1	0
1776Katherine	03/11/2020 21:03	@realDonaldTrump God we ask you for a huge LANDSLIDE!!!	0	1
1776Katherine	03/11/2020 01:41	@realDonaldTrump AMERICA'S CHOICE IS #Trump2020 https://t.co/otP4H1scdJ	0	1
1776Katherine	03/11/2020 21:06	@RepsForBiden @realDonaldTrump What a lie .	-1	0
1776Katherine	03/11/2020 21:06	@realDonaldTrump Biden saved nothing!	-1	0
1776Katherine	03/11/2020 21:12	@AndyOstroy @realDonaldTrump #Trump2020Lanslide	0	1
1776Katherine	03/11/2020 01:41	@realDonaldTrump Ladies and gentlemen, Trump is Triumph @realDonaldTrump and tomorrow the trumpets will sound announcing	0	1
1776Katherine	03/11/2020 23:02	@realDonaldTrump I stood in line for 2 hours fir the best PRO America @POTUS ever! Re elect our great AMERICAN @realDonaldTrur	0	1
1776Katherine	03/11/2020 23:09	Love our @POTUS FOUR MORE YEARS! Make it happen AMERICA!!!	0	1
1776Katherine	04/11/2020 01:08	Come on Nevada! Get out and vote. @realDonaldTrump was down four points by registration after Election Day in 2016 and lost by t	0	1

User class prediction probability

```
predict_probability('1776Katherine')
```

	Class	Probability
0	supporting Trump, Biden neutral	0.4252
1	supporting Biden, Trump negative	0.2322
2	Biden neutral, Trump negative	0.1361
3	neutral	0.0803
4	supporting Trump, Biden negative	0.0714
5	supporting Biden, Trump neutral	0.0299
6	Trump neutral, Biden negative	0.0249

Figure 36 – Model error interpretation on individual user #1

We can see that the most probable class for this user was “supporting Trump, Biden neutral” as per the original tweets the actual sentiment was negative towards Biden.
However, we can see that the second most probable class was with negative sentiment towards Trump.
This can be assigned to specific words in tweets with strong negative sentiment, such as “lie” and “...was down...”, and the fact that both candidates are mentioned in those tweets.
Once again, we see that the even though the context is lost when merging the tweets, the model does a good job in recognizing the sentiment and the candidate in focus.

User

	user	Correct class	Correct classmodel probability	Predicted model class	Predicted model class probability	Correct class probability rank	Correct class in the top 3 predictions of the model
36	HamidYosefi00	supporting Trump, Biden negative	0.0501	supporting Trump, Biden neutral	0.629721	4	No

User tweets

usu	created_at	text	B	T
HamidYosefi00	05/11/2020 06:07	Do not leave the president of the hearts alone...@realDonaldTrump #GivebackTrumpvotes https://t.co/nu0nD688xl	0	1
HamidYosefi00	04/11/2020 20:18	@realDonaldTrump You must be next president, stay strong 💪#GivebackTrumpvotes	0	1
HamidYosefi00	04/11/2020 22:29	the bloodthirsty dictator of Iran, who has killed more than 3,000 people in the past year and does not even allow his citizens to use t	-1	1
HamidYosefi00	05/11/2020 03:13	Hold the line!#GivebackTrumpvotes	0	1
HamidYosefi00	04/11/2020 20:19	Yeah you are going great Sir! #GivebackTrumpvotes #voterfruad	0	1
HamidYosefi00	04/11/2020 22:26	mr @realDonaldTrump the iranian pepole never forgot you for everything we stand youAnd we love you♡#GivebackTrumpvotes	0	1
HamidYosefi00	05/11/2020 00:13	From iranian people We are proud of you #GivebackTrumpvotes @realDonaldTrump	0	1
HamidYosefi00	05/11/2020 04:40	@bamdad_azad @Rosshanak @realDonaldTrump Only for this 👇#GivebackTrumpvotes https://t.co/8dR6xlR9c8	0	1
HamidYosefi00	04/11/2020 20:51	Dear President Trump, We, the people of Iran, are by your side. When everyone was silent, you were with us. You are victorious. Long	0	1
HamidYosefi00	04/11/2020 21:22	We wont leave you alone mr.president ♡UMIR#GivebackTrumpvotes #Trump2020 #MAGA2020 @realDonaldTrump https://t.co/yzCJi	0	1

User class prediction probability

```
predict_probability('HamidYosefi00')
```

	Class	Probability
0	supporting Trump, Biden neutral	0.6297
1	supporting Biden, Trump negative	0.1206
2	Biden neutral, Trump negative	0.1110
3	supporting Trump, Biden negative	0.0501
4	neutral	0.0457
5	supporting Biden, Trump neutral	0.0222
6	Trump neutral, Biden negative	0.0208

Figure 37 - Model error interpretation on individual user #2

We can see that the most probable class for this user was “supporting Trump, Biden neutral” again, as per the original tweets the actual sentiment was negative towards Biden.

This can be assigned to a very vague tweet with strong negative sentiment which was marked as Biden negative and again, the fact that both candidates are mentioned in that tweet.

As well, we can see that the majority of tweets (9 out of 10) are actually Biden neutral.

User

	user	Correct class	Correct classmodel probability	Predicted model class	Predicted model class probability	Correct class probability rank	Correct class in the top 3 predictions of the model
2	AjeebInsaan15	supporting Trump, Biden negative	0.1298	Biden neutral, Trump negative	0.256407	5	No

User tweets

B	C	D	E	F
usu	created_at	text	B	T
AjeebInsaan15	03/11/2020 05:48	The countdown has begun...it's almost #ElectionDayand @realDonaldTrump WILL BE DONE!https://t.co/2ZS0NQO8w6	0	-1
AjeebInsaan15	03/11/2020 06:22	Let the record reflect that 5 citizens of #DixvilleNotch NH were the first voters to be recorded on this Election Day telling @realDonalc	0	-1
AjeebInsaan15	03/11/2020 07:18	This 74 year old man is speaking to America at 1:10 am at his 10th rally in two days. Get out and vote for @realDonaldTrump...UMUM	0	1
AjeebInsaan15	03/11/2020 08:28	Thank you Paris. Keep up the GREAT work!	0	0
AjeebInsaan15	03/11/2020 18:16	PENNSYLVANIA — As you head to polls, remember this: @realDonaldTrump has brought jobs back to America through fairer trade dea	-1	1
AjeebInsaan15	03/11/2020 16:43	Thank you to our 2.6 MILLION volunteers!You have made over 185 MILLION phone calls and door knocks for @realDonaldTrump and	0	1
AjeebInsaan15	03/11/2020 15:26	ATTENTION MICHIGAN! All eyes are on our state — let's make sure we re-elect @realDonaldTrump and send @JohnJamesMI to the Ser	0	1
AjeebInsaan15	03/11/2020 14:14	Our economic recovery is on the ballot.Our energy industry is on the ballot.Biden promised to raise taxes and stop fossil fuels. Vote	-1	1
AjeebInsaan15	03/11/2020 17:55	#Trump says he built greatest economy, which was 'horribly' interrupted by 'plague' from China #USAElections2020 https://t.co/njDC	0	1

User class prediction probability

```
predict_probability('AjeebInsaan15')
```

	Class	Probability
0	Biden neutral, Trump negative	0.2564
1	supporting Trump, Biden neutral	0.2033
2	supporting Biden, Trump negative	0.1658
3	neutral	0.1576
4	supporting Trump, Biden negative	0.1298
5	supporting Biden, Trump neutral	0.0636
6	Trump neutral, Biden negative	0.0235

Figure 38 - Model error interpretation on individual user #3

In this case, we have a very ambiguous user where we have both negative and positive sentiments toward Trump, and some negative sentiment toward Biden.

The model classified negative sentiment and support toward Trump very closely with 0.25 and 0.20 probability, which probably was very hard to differentiate once all tweets were merged

6. Conclusions

Due to the unprecedented rise of data content over the last decade an opportunity for data-based analysis has become a norm in the modern world. As social media becomes increasingly central to young people's everyday lives offering a direct no-filter way of expressing our thoughts on these platforms, it is important to understand how and in what measure can a platform, like Twitter, help analyse such a complex geo-political happening as US elections 2020.

Even though the composition of social media users can never be considered the same as a real demographic, it still offers a very important insight into public opinion and influence.

In this work, we performed a data analysis on tweets about the 2020 US Presidential election for candidates Donald Trump and Joe Biden. The two main objectives were 1) to gain insights into general sentiments and trends for both candidates and 2) to classify the users into 9 classes depending on their general message from tweets.

We started with 2 datasets, one containing individual tweets from unique users and the other containing multiple tweets from unique users. All tweets were manually classified with appropriate sentiment regarding Trump and Biden and the 9 classes, into which the user's tweets were grouped, were determined by the combination of those sentiments. For the dataset with multiple tweets from the same user, we merged all their tweets into one conjoint text and the sentiment for the user regarding Trump and Biden was drawn from the average sentiment from all their tweets and then rounded up. Immediately we saw that we had very much imbalanced classes for both datasets and we would have to use partitioning techniques such as SMOTE to manage the distribution for the prediction algorithms to achieve satisfying results.

With the data sample we had, neutral tweets were a large majority, but Trump had more "negative" and "positive" sentiment tweets than Biden. Twitter is famously the preferred social media platform of the now-ex US president Donald Trump and perhaps that might have something to do with the higher number of tweets that had some sentiment toward the candidate.

An interesting insight was that even though we only had tweets up until the election day, November the 3rd, we had Trump supporters' tweets trending with words like "give back Trump votes", "Michigan rigged" or "rigged election" even before the results were official or near conclusion.

We could see on average that tweets regarding Biden included much more individuals than Trump based tweets both "supportive" and "negative" tweets. A strong inclusion of Kamala Harris and Barack Obama in the "supportive" tweets and Hilary Clinton and Hunter Biden, the son of now-President Joe Biden were the most used individuals in tweets with "negative" sentiment toward Biden.

The peak of the tweet count happened on the election day where, again, tweets regarding Trump were in bigger number than tweets regarding Biden. Supportive tweets were the majority for Trump, where for Biden was the opposite.

After the exploratory analysis we constructed multiple predictive algorithms to get the probability of the user belonging to one of the 9 classes previously assigned.

A TF-IDF vectorization was performed on the now cleaned and lemmatized tweets, transforming them to a numeric (vectorized) format that the Machine Learning models can use for training and testing. We applied and optimized these models first on the larger, individual users' dataset, which we later used as a training dataset for our multiple tweets from unique user dataset where we faced a real challenge.

As the tweets were merged for each unique user, it was easy to lose its context as well as the sentiment for one candidate to be mixed up for another. Therefore, we opted to present the probability of the user to belong to one of these classes instead of just measuring the model accuracy on the highest probability prediction.
We constructed a dataset where we ranked the model's prediction and compared them to the actual class that the user was assigned to, as a result we had a XGB prediction model with 63% of users where the actual class assigned to the user was in the top 3 classes predicted by the model. Given that we had a multiclass classification problem with a small to medium dataset, we believe this presented a good result and the model was predicting rather well.

The error analysis gave us a valuable insight into the model performance. We saw that indeed the model often confused the sentiment showed for one candidate to another, regularly when both candidates were mentioned in a tweet with very strong "positive" or "negative" connotation.

In regard to future work, there are numerous ways of the improvement of the analysis with larger datasets giving us a better sample, different methods for text analysis such as "Bag of Words" or a more complex lemmatization technique using a larger n-gram parameter to improve the understandability of the textual context inside the tweet. We can approach the problem differently and perhaps more efficiently by building more complex prediction algorithms with Neural Networks or a more accurate hyperparameter optimization for our models with stronger computational power.

Construction of such a multiclass classification model for user group assignation can prove to be very useful. Specifically in cases where we might have the presence of polarity or communities in the subject, such as we had here in the 2020 US Elections. Models such as these can be used to monitor the trends and the growing groups of individuals present on the social media platforms.

The process of ongoing world digitalization and the increasing social media presence will produce increasingly more data analysis and machine learning applications across a wide range of areas and domains. These platforms, such as Twitter, offered us a minable opinionated individual data for the first time in the history and as a result we have an incredible opportunity to analyse and target specific groups of individuals based solely on the content uploaded which is presumed to represent their thoughts, beliefs, and opinions.

7. Bibliography

[1] Morales, J. M. R. (2011). Ciudadanía digital: Una introducción a un nuevo concepto de ciudadano. Editorial UOC.

[2] Dey, P., Kothari, P. K., & Nath, S. (2019, January). The social network effect on surprise in elections. In Proceedings of the ACM India Joint International Conference on Data Science and Management of Data (pp. 1-9).

[3] Ceron, A., Curini, L., & Iacus, S. M. (2015). Using Sentiment Analysis to Monitor Electoral Campaigns: Method Matters—Evidence From the United States and Italy. Social Science Computer Review, 33(1), 3–20. https://doi.org/10.1177/0894439314521983

[4] Patil, A. P., Doshi, D., Dalsaniya, D., & Rashmi, B. S. (2017, September). Applying Machine Learning Techniques for Sentiment Analysis in the Case Study of Indian Politics. In International Symposium on Signal Processing and Intelligent Recognition Systems (pp. 351-358). Springer, Cham.

[5] 'Cambridge Analytica CEO Claims Influence on U.S. Election, Facebook Questioned'. Reuters, 20 March 2018, sec. Media and Telecoms. https://www.reuters.com/article/us-facebook-cambridge-analytica-idUSKBN1GW1SG.

[6] Detrow, Scott. 'What Did Cambridge Analytica Do During The 2016 Election?' NPR, 20 March 2018, sec. Politics. https://www.npr.org/2018/03/20/595338116/what-did-cambridge-analytica-do-during-the-2016-election.

[7] Witten, I.H., Frank, E., Hall, M.A., Pal, C.J. and DATA, M., 2005. Practical machine learning tools and techniques. In DATA MINING (Vol. 2, p. 4).

[8] Géron, A., 2019. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media.

[9] What is Natural Language Processing? An Introduction to NLP [WWW Document], n.d., SearchEnterpriseAI. URL https://searchenterpriseai.techtarget.com/definition/natural-language-processing-NLP)

[10] Ott, B. L. (2017). The age of Twitter: Donald J. Trump and the politics of debasement. Critical studies in media communication, 34(1), 59-68.

[11] Yaqub, U., Sharma, N., Pabreja, R., Chun, S. A., Atluri, V., & Vaidya, J. (2020). Location-based Sentiment Analyses and Visualization of Twitter Election Data. Digit

[12] Baker, S.R., Baksy, A., Bloom, N., Davis, S.J., Rodden, J.A., 2020. Elections, Political Polarization, and Economic Uncertainty, NBER working paper series. National Bureau of Economic Research, Cambridge, Mass.

[13] Harris, M.D., 1985. Introduction to natural language processing. Reston Publishing Co.

[14] Bitext. 'What Is the Difference between Stemming and Lemmatization?' https://blog.bitext.com/what-is-the-difference-between-stemming-and-lemmatization/.

[15] Balakrishnan, V. and Lloyd-Yemoh, E., 2014. Stemming and lemmatization: a comparison of retrieval performances.

[16] Vallantin, Lima. 'Why Is Removing Stop Words Not Always a Good Idea'. Medium (blog), 15 June 2020. https://medium.com/@limavallantin/why-is-removing-stop-words-not-always-a-good-idea-c8d35bd77214.

[17] Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P., 2002. SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, pp.321-357.