Sentiment Analysis on COVID-19 Vaccines in Ireland using Support Vector Machine

2021

by

Karla Aniela Cepeda Zapata
Under the supervision of
Dr. Rajesh Jaiswal

School of Informatics and Creative Arts
Department of Computing Science and Mathematics

Acknowledgements

All access and data used for this project was provided by Twitter, Inc. No fees were charged, as their product tracker *Academic Research* is aimed to advance research objectives with public data on nearly any topic. The data collected from this product tracker is strictly for non-commercial purposes.

I would like to thank the head of the Department of Computing Science and Mathematics, Dr. Fiona Lawless, to support me establishing my identity to Twitter, Inc. to gain access to the Twitter API product.

I would especially like to thank Dr. Rajesh Jaiswal, the Director of Data Analytics program in Dundalk Institute of Technology. As my lecturer and supervisor, he has taught me more than I could ever give him credit for here. He has shown me, by his example, what a good scientist (and person) should be.

I am grateful to all of the lectures from the Data Analytics program, Dr. Siobhan Connolly Kernan, Dr. Jack McDonnell, Dr. Peadar Grant and Dr. Caroline Sheedy, whom provided me extensive professional knowledge and taught me a great deal about data science skills during lectures, laboratories and projects.

This work is self-funded, and I would like to thank Dundalk Institute of Technology (DkIT) for facilitate the MSc in Data Analytics program that has helped me to work on this project. Additionally, I am grateful by the access provided to online courses on DataCamp platform to enhance data science skills.

I would like to thank my parents, Martha Leticia Zapata Vaquera and Luis Carlos Cepeda Ochoa, whose love and guidance are with me in whatever I pursue. They are the ultimate role models.

I wish to thank Daniel Traynor, who provide unending inspiration, support and love.

Declaration

“I hereby declare that the work described in this project is, except where otherwise stated, entirely my own work and has not been submitted as part of any degree at this or any other Institute/University”

Signed Karla Cepeda

Name Karla Aniela Cepeda Zapata

Date May 29, 2021

Dedication

To my mother, Martha Leticia, and my father, Luis Carlos. Without their love, support and encouragement I would not be here.

To Daniel, for all his love, support and inspiration.

Abstract

This document describes the process of conducting a Sentiment Analysis on the Covid-19 vaccines in Ireland from the 1st of January 2020 to the 13th of August 2021. The source of data chosen is the social media Twitter since being more used to share opinions, whereas other social media platforms such as Facebook are more used to connect with others and shared media objects such as pictures or videos. Irish tweets were collected with Python via Twitter API by sending queries to the "Full-archive search" endpoint that contained the keywords "covid" and "vaccine" and any synonym or related word. The collection stage generated two datasets: global tweets (for labeling and modeling stages, excluding the place Ireland) and Irish tweets (for analysis stage). In order to gather Irish tweets, the operator "place:ie" (i.e., Ireland) was included. Other queries also included Irish users related to media, government, and health departments to increase observations. Then, the lexicon and rule-based tool VADER was used to classify the global tweets into positive, negative, and neutral for the modeling stage. Mainly, this lexicon tool was chosen as previous studies showed a better accuracy score on social media texts. The Machine Learning (ML) technique utilized for the modeling stage was Support Vector Machine (SVM), a supervised learning model for classification and regression applications. This algorithm has been applied in the past displaying a good accuracy score for text classification. The model was built using the package scikit-learn in Python. After labeling the Irish tweets using the model, on average, the data showed that from January 2020 to February 2021, there were positive feelings towards the Covid-19 vaccine in Ireland. However, this feeling changed after February 2021 to a *negative* feeling, and the recent feeling (the second week of August 2021, latest data collected) remained *negative*.

The code designed for this project is available on GitHub following the link: *https://github.com/karla-cepeda/Dissertation*. A dashboard was deployed to display the results on *https://sentimentanalysis-c19v-ie.herokuapp.com*. Additionally, a screen-cast was recorded to show the Python code's functionality and structure, that is available on *https://web.microsoftstream.com/video/ead11ffc-78ea-4394-97c3-db6ad07d9153* (please, see the description of the video before playing).

Abbreviations

AI Artificial Intelligence. 36

BoW Bag-of-words. 57

Covid-19 Coronavirus disease 2019. iv, ix–xi, 1–3, 7–10, 12–16, 41, 43, 55, 67, 82, 86–89, 106, 111

CRISP-DM Cross-Industry Standard Process for Data Mining. 2, 4, 86

DB database. 4, 5, 48, 64, 68, 80, 86, 88, 89

DBMS Database Management System. 4, 48, 64

DkIT Dundalk Institute of Technology. i, 63

EMA European Medicines Agency. 12

EUA Emergency Use Authorization. 12

FAO Food and Agriculture Organization of the United Nations. 9

FDA Food and Drug Administration. 12

GDPR General Data Protection Regulation. 63

HPSC Health Protection Surveillance Centre. 15, 111

HSE Health Service Executive. 15–17, 43, 54, 59, 111

HTTP Hypertext Transfer Protocol. 20, 21

k-NN k-Nearest Neighbor. 34

LIWC The Linguistic Inquiry and Word Count. 39

ME Maximum Entropy. vii, 30, 32, 34

ML Machine Learning. iv, vii, 1–5, 27, 28, 30, 31, 33–35, 39, 48, 68, 86, 106

MNB Multinomial Naive Bayes. 30, 34

NB Naive Bayes. vii, 30, 31, 34

NN Neural Network. vii, x, 30, 32–35

NPL Natural Processing Language. 28, 32, 36, 39, 40, 87, 106

OIE Organisation for Animal Health. 9

RNN Recurrent Neural Network. 34, 35

SARS-Cov-2 Severe acute respiratory syndrome coronavirus 2. 9

SVM Support Vector Machine. i, iv, vii, 29–31, 34, 35, 68, 69, 71, 73, 74, 86, 87

TF Term Frequency. 68

TFIDF Term Frequency-Inverse Document Frequency. 68, 86

VADER Valence Aware Dictionary for sEntiment Reasoning. iv, 35, 39, 66, 67, 86

WHO World Health Organization. 9, 10, 12, 16, 116

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Although many people strongly agreeing vaccines are effective[1], mixed points of view on vaccines are derivative by different factors and events. A remarkable episode occurred in 1998, where Andrew Wakefield and 12 colleagues proposed that the MMR (measles, mumps, and rubella) vaccine was linked to autism disorders in children. Despite this being found false, this particular paper affected the sentiment of the people back in that time, resulting in massive measles outbreaks in countries like the United States of America, the United Kingdom, and Japan (Tss & Andrade 2011). Even after decades, this is still affecting people's opinions on vaccines. In 2016, the Statista Research Department conducted a survey in the United States of America in which it was found that one of the most common reasons for the hesitation of vaccination was the "*fear of connection to autism spectrum disorder*." Additionally, other reasons found to refuse vaccines were: concerns about side effects, being ineffective or safe, fear of getting sick from the vaccine, being afraid of needles, and religious or political beliefs. Having said that, it is essential to identify the sentiment of the public along with the Covid-19 vaccine roll-out in Ireland because these feelings may be affected by this new virus, which has brought many changes worldwide. As there is no drug to treat the new virus, lockdowns and restrictions were implemented to contain the spread of the virus, causing uncertainty and anxiety in society. Historically, vaccines have been effective against several diseases, saving millions of lives. Hence, a massive vaccination program has been seen as a solution to return to a *normal* life. Nonetheless, it is assumed that the sentiment on these vaccines is strongly affected, mainly because of their fast development and approval of vaccines from four pharmaceutical companies (i.e., BioNTech/Pfizer, Oxford/AstraZeneca, Moderna, and Johnson&Johnson). In order to find and analyze this sentiment, a set of tweets was collected via Twitter API, and a Machine Learning algorithm was trained to classify tweets as positive, negative, and neutral with the *scikit-learn* package in Python. Finally, after labeling the tweets, a dashboard was designed to display the results. This methodology will be discussed in more detail in further sections of this thesis.

[1] According to John Elfien (2018), around 63% of people worldwide agree vaccines are effective and safe.

This project aims to identify the general sentiment on the Covid-19 vaccines in Ireland, collecting tweets from *1 January 2020* to *13 August 2021*. A Machine Learning (ML) algorithm were trained and tested to classify tweets into positive, negative, or neutral opinions. Knowledge from Programming, Data Architecture, Ethics, Statistics, and Machine Learning modules from MSc in Data Analytics was used throughout the project. Table 1.1 summarizes the scope[2] of the research and the goals. As the Sentiment Analysis is a new topic, a DataCamp courses are included in the project goals.

The relevance of the project in the Data Science field relies on the performance of Sentiment Analysis on data collected from Twitter to understand the opinion on the Covid-19 vaccines in the Republic of Ireland, using the Cross-Industry Standard Process for Data Mining (CRISP-DM) life-cycle methodology to implement this project. Additionally, this topic was chosen (i.e., Covid-19 and vaccines) due to its relevance since this is an ongoing worldwide event that concerns not just researchers but also the public.

Appendix D.1 contains the proposed analysis for this project presented in the interim report on 15 June 2021.

[2] Project Goals were sorted by priority.

Research Questions	Project Goals
• How to collect data from Twitter API? • What is the ... – day – week – month – quarter ... with the largest and lowest (excluding zero) number of Irish tweets? • How to label tweets by positive, negative, or neutral opinion? • How has the general opinion been changing over time? • What is the general perception of the public on... – government – healthcare system – mass media ... stand on vaccines in Ireland? • What is the overall feeling on the Covid-19 vaccine in Ireland? • What is the recent feeling on the Covid-19 vaccine in Ireland?	• Get access to Twitter product Twitter API. • Collect tweets related to covid vaccine and any synonym or related word. • Research on critical dates related to Covid-19 and vaccines in Ireland and worldwide. • Take Natural Language Processing courses on DataCamp online platform: – Introduction to natural language processing in Python. *https://www.datacamp.com/courses/introduction-to-natural-language-processing-in-python* – Natural Language Processing in Python. *https://www.datacamp.com/tracks/natural-language-processing-in-python* – Advanced NLP with spacy. *https://www.datacamp.com/courses/advanced-nlp-with-spacy* • Perform a text analysis on tweets using word cloud for the most frequent words. • Perform analysis on tweets with one-word length. • Research on labeling process. • Labeling tweets using two methods: – Employing an online sentiment analyzer tool such as Azure or Monkey Learn. – Building a model using Machine Learning tools. • Understand the opinion over the time for each vaccine available in the Republic of Ireland (i.e., Pfizer, Moderna, and AstraZeneca vaccine). • Understand if the Irish mass media, Irish government, and Irish healthcare system accounts on Twitter are showing a feeling when spreading news regarding vaccines. • Analysis on tweets with one-word length (not including stop words) to discard those not significant • Create a process to retrieve daily tweets.

Table 1.1: Research questions.

Python was the programming language chosen for this project. In 2020, Kaggle, one of the largest data science community platforms, conducted a worldwide survey showing that the most used programming language in the data science field is Python due to its popularity and easy learning curve (Kaggle 2020). Additionally, numerous libraries are available for data manipulation, visualization, and ML applications for this language (IBM Cloud Team, IBM Cloud 2021). Related to database (DB), the Database Management System (DBMS) chosen was MySQL for this project. Twitter uses MySQL primarily for the storage of Tweets and Users (Twitter Inc. 2017). Moreover, Murazza and Nurwidyantoro (2016) compared DB in a near real-time Twitter ware-house system, where MySQL has performed better when reading queries comparing to Casandra and PostgreSQL. Table 1.2 shows a summary of the technology implemented in this project.

Core technology

• Programming Language: **Python**
  – IDE: Spyder, Visual Studio Code and Jupyter Notebook
  – Libraries:
  * pandas
  * numpy
  * requests
  * sklearn
  * nltk
  * mysql_connection
  * dash
  * maltplotli and plotly
• **Twitter API** (application previously approved by the Twitter Dev Team).
• For storage: **MySQL** database.
• For visualization of the data: **Power BI**.
• For version control system: **GitHub**.

Table 1.2: Core technology used in project.

This project took roughly three months to complete. The framework was based on the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology, providing guidance and an overview of data mining projects, including data science projects (IBM 2021). Figure 1.1 shows the whole life cycle of this project, made up of seven stages:

- **Data Collection**. The data was collected from the social media Twitter. This stage included coding a script to connect to the API. Additionally, batches were designed to build queries sent to the "Full-archive search" endpoint to download Irish tweets (i.e., tweets that have been posted by residents in Ireland). A previous trial was performed in March 2021, collecting fewer than 10,000 observations. A second trail was

performed in May 2021, collecting around 30,000. Therefore, an additional batch was designed to collect global tweets for modeling purposes, as it was aspired to train and test the ML algorithm with at least 100,000 observations. However, in June and July 2021 the number of Irish tweets increased unexpectedly as a consequence of modifications to the Python code. Even though the number increased, the plan of using global tweets for modeling continued.

- **Data Preparation**. This stage involved cleaning and normalization tasks on tweets collected, including removing mentions, hashtags, and punctuation marks. Additionally, the cleaned and normalized data were stored into the DB, including the insertion of users, place, and referenced tweets information.

- **Data Exploration**. Cleaned and normalized data were plotted to describe Irish tweets and general features[3]. A brief description of global tweets was performed as this is a second dataset mainly used for the modeling stage.

- **Labeling**. As the model designed was a supervised learning subcategory of ML, it was required to have a labeled dataset. This stage was included not to manually classify the tweets as it was expected to collect more than 100,000 observations. A research process was conducted on how to classify tweets into positive, negative, and neutral sentiments.

- **Modeling**. A Machine Learning algorithm was trained and tested with global tweets already labeled. Before this, a research process was performed to select the algorithm to implement.

- **Evaluation**. The evaluation process is performed by examining accuracy scores. Additionally, different plots are used to identify signs of over-fitting/under-fitting. If the accuracy score of train and test datasets are deficient, or there are signs of over-fitting/under-fitting, tuning hyper-parameters are performed to improve the score. Once the evaluation is completed, Irish tweets will be labeled using the model created.

- **Deployment**. A dashboard is designed to visualize the labeled tweets. It is planned to design a line plot to show the sentiment over time, an indicator to display the global sentiment, and another indicator to display the last sentiment stored on the database. Additionally, a database is set up in a remote location, a daily tweet collector is coded, and a process to migrate the local tweets to the remote database is designed.

[3]No sentiment analysis would be performed in this stage yet.

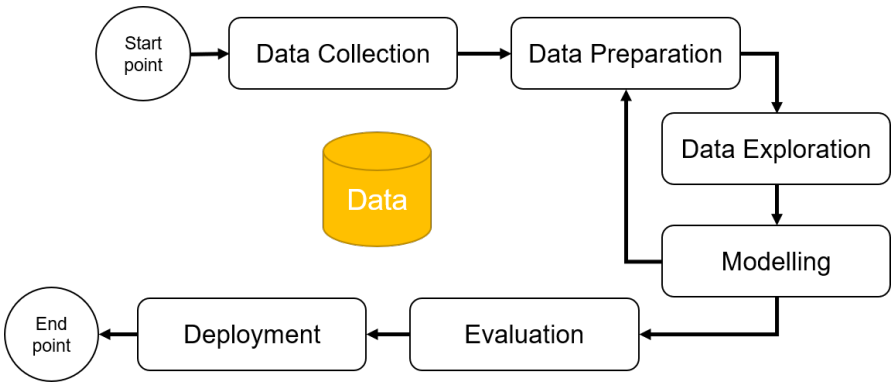


Figure 1.1: Life cycle of the project.

The structure of the report starts off Chapter 2 (Literature Review) that describes the most relevant topics related to the research questions shown in Table 1.1. Subsequently, Chapter 3 (Exploration of the Data) describes all processes performed to collect and clean tweets to get a final dataset for analysis. Chapter 4 (Design and Implementation) explains the labeling process and design of the based model for training and test of global tweets. Chapter 5 (Parameter Tuning, Evaluation, and Testing) explains the process to assess the based model. Chapter 6 (Results) discusses the sentiment of the Irish tweets over time in general, recently, and by usernames and vaccines. Chapter 7 (Conclusion) discusses whether the research questions were addressed and future work. Finally, Chapter 8 (Deployment) explains the structure of the dashboard that shows the results discussed in Chapter 6.

Chapter 2

Literature Review

2.1 Related work

Several studies have been conducted to explore the sentiment of the public on different topics and domains. In the Covid-19 domain, Gupta et al. (2021) performed an analysis on the global reactions utilizing Twitter data. They collected tweets from 28 January 2020 to 1 January 2021, obtaining up to 132 million tweets from 20 million unique users. Their results indicate that the majority of the tweets were “negative” or “very negative”, adding up to 59.4% of the total of tweets, whereas "positive" or "very positive" tweets were 25% of the total. They also classified the tweets according to an emotion, where "anger" has the largest proportion of tweets.

Raghupathi et al. (2020) have implemented a Sentiment Analysis on "vaccination" using Twitter data. They reported that 43.3% of the tweets were classified as "negative", 40.4% were "positive" tweets, and 16.3% were "neutral", noticing the number of "negative" tweets was slightly higher compared to "positive" tweets. In the Covid-19 vaccine domain, Chen Lyu et al. (2021) carried out a Sentiment Analysis on Covid-19 vaccines, collecting tweets from 11 March 2020 to 31 January 2021. Despite the fluctuation, they found out that the sentiment was increasingly "positive". In an emotion-grained analysis, "trust" was the most predominant emotion. In another study on Sentiment Analysis on vaccines, Yousefinaghani et al. (2021) observed a large proportion of tweets classified as neutral. In general, they concluded that "positive" tweets dominate almost all weeks. Although Raghupathi et al. (2020) found a large proportion of negative sentiment, it seems that there is a positive feeling towards Covid-19 vaccines as a sign of "trust" emotion.

2.2 A new coronavirus

Nobody[1] foresaw its impact when a novel virus was discovered on the 31st December 2019 in Wuhan, Hubei province, China. After multiple cold-like illness cases connected to the Huanan Seafood Market, this was closed down on the 1st of January 2020. The *Chinese Health Authorities* identified this virus as a new strain of *Coronaviridae* (Zhang et al. 2020),

[1] Some fragments on this paper were taken from the 2nd Continuous Assessment, Ethics in Data Analytics module.

a large family of "*positive-sens, single-standard RNA viruses that belongs to the Nidovirales order*", and related severe acute respiratory syndrome (SARS) (Fauci et al. 2020). The *Coronaviridae* could be isolated from different species such as birds, camels, mice, dogs, and cats (Hassan et al. 2020). Furthermore, this type of virus can affect human beings, from which seven different coronaviruses had been detected before (UK Reserach and Innovation 2020). Evidence has shown the new coronavirus crossed species to infect humans as it had been found on *bats* (more precisely on the *Rhinolophus* sub-species) as the genome-wide nucleotide sequence is 96% identical to these mammals (World Health Organization 2020*b*). Nevertheless, human-to-human was considered the main way of transmission despite the variety in exposure history from confirmed cases. Symptomatic cases are the main sources of infection. However, people who remain asymptomatic could infect other individuals as well. Transmission among humans occurs when (World Health Organization 2021*b*):

- An infected person coughs/sneezes/speaks spreading droplets that contain the virus, closing contact among individuals within a one-meter distance.
- The virus could be spread in closed and poorly ventilated spaces due to a high concentration of aerosols, since these travel on the air within a one-meter distance.
- People may be infected when touching contaminated surfaces and then touching their face.

The symptoms that infected people develop are enlisted in the table 2.1, classifying them as "common", "sometimes", "unusual", and "never". Since there is notable overlap among symptoms of Covid-19, *Flu* and *Cold*, this table includes a comparison among symptoms and these diseases. These could appear between 2 to 14 days, and not all infected individuals may have all of these symptoms (Health Service Executive 2021) (Li et al. 2020).

Symptom	**COVID-19**	**Flu**	**Cold**
Fatigue	Common	Common	Common
Aches and pains	Common	Common	Common
Cough	Common (dry)	Common (dry)	Sometimes
Fever	Common	Common	Unusual
Lost or changed sense of smell / taste	Common	Unusual	Unusual
Shortness of breath	Common	Never	Never
Sore throat	Sometimes	Sometimes	Common
Runny nose or nasal congestion	Sometimes	Sometimes	Common
Headaches	Sometimes	Common	Unusual
Nausea / vomiting	Unusual	Sometimes	Never
Diarrhoea	Unusual	Sometimes (in children)	Never
Sneezing	Never	Never	Common

Table 2.1: Covid-19 symptoms and comparison among other diseases.

In just a month, the virus spread rapidly in all provinces in China. Due to the Spring Festival, a national holiday that takes place every year on January (timeanddate.com n.d.), roughly 680 cases were exported outside China (more than 25 countries). So far, the countries that have been more affected are The United States of America with over 32 million confirmed cases; India with over 26 million confirmed cases; Brazil with over 16 million confirmed cases; and France with over 5million confirmed cases [2]. Figure 2.1[3] shows a graph from January 2020 to the 25th of May 2021, when the highest number of cases confirmed worldwide happened in April 2021 (World Health Organization 2021*c*).

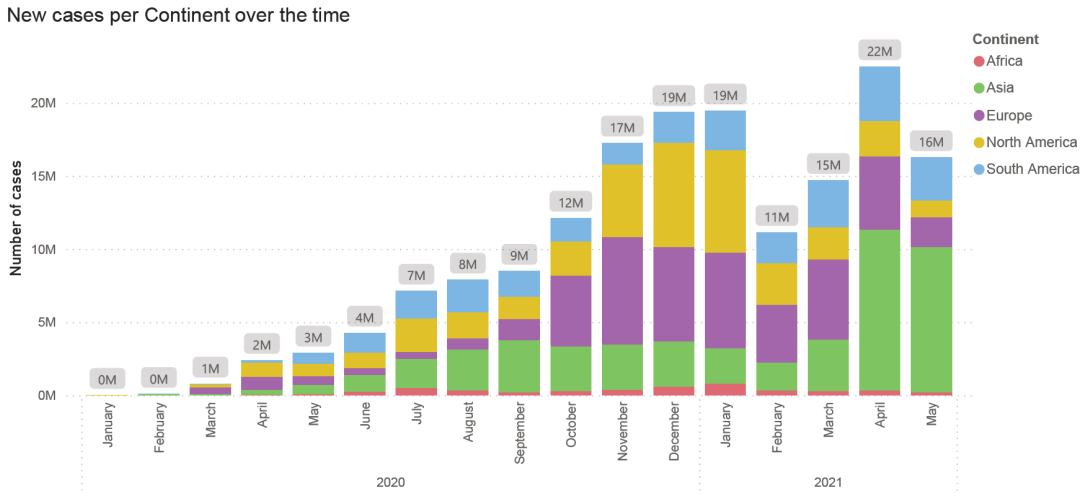


Figure 2.1: Confirmed COVID-19 Cases in Ireland over the time.

By following best practices and in collaboration with the Organisation for Animal Health (OIE) and Food and Agriculture Organization of the United Nations (FAO), the World Health Organization (WHO) had named the virus as *Severe acute respiratory syndrome coronavirus 2* (SARS-Cov-2) and the disease as *Coronavirus disease 2019* (Covid-19) (World Health Organization 2020*b*). The virus and the disease caused often have different names. The name of a virus is based on the genetic structure whereas the name of the disease is determined to allow discussion on prevention, spread, severity, and treatment. As explained by WHO, the term SARS-Cov-2 is not often used when communicating with the public to halt "*unintended consequences*" created by fear on the population. Therefore, WHO refers to the virus as "*the virus responsible for Covid-19*" or "*the Covid-19 virus*" (World Health Organization 2020*a*).

In a media briefing, the WHO Director-General, Dr. Tedros Adhanom Ghebreyesus, expressed their "*deeply concerned [...] by the alarming level of spread and severity, and by the levels of inaction*" and the expectation "*to see the number of cases, the number of deaths, and the number of affected countries climb even higher*". Hence, on the 11th March 2020, WHO had declared the novel coronavirus as a *global pandemic* (World Health Organization 2020*c*).

[2] Date of information: 25 May 2021

[3] Source: Our World in Data. Created on Power BI, © 2021 Microsoft.

According to WHO, to reduce the risk of infection it is suggested to follow these basic precautions[4] (World Health Organization 2021*a*):

- Wear a mask made up of three layers. Strongly recommended when it is not possible to keep at least a one-meter distance away from others.
- Avoid spaces that are closed, crowded, or involve close contact (WHO called this the 3Cs).
- In public spaces, avoid touching surfaces as these could be contaminated by infected people. Recommended disinfecting spaces regularly.
- Frequently clean hands with soap and water or rub hands with an antibacterial gel.
- To cough, avoid using hands. Instead, use one arm or a tissue. If tissue was used, drop it straight away to the bin.

2.3 A hope to control the pandemic

As the number of confirmed cases and the number of deaths were increasing dramatically, leading to critical challenges for the public healthcare system worldwide, the search for a drug against the Covid-19 virus failed to control the pandemic. For many, especially in the scientific field, the only way to solve this is by the development of a vaccine for immunization. Therefore, vigorous research to develop a preventive vaccine against the virus that causes the Covid-19 is the best approach to hold the pandemic (Fauci et al. 2020). Expectations for effective vaccines are high as WHO has listed more than 200 vaccines under development (Haynes et al. 2020).

Vaccines are "*biologics that provide active adaptive immunity against specific diseases*" (Kashte et al. 2021). This a product that is introduced into the human body via the mouth, injected, or through the nose to stimulate the immune system and create immunity against specific disease (Centers for Disease Control and Prevention - USA 2018). The development of a vaccine involves utilizing the same germs that cause disease, however, these have been killed/weakened (Centers for Disease Control and Prevention - USA 2012). There are different types of vaccines which are mentioned in the list below (Khuroo et al. 2020, Kaur & Gupta 2020, Dai et al. 2001)[5]:

- **Live attenuated vaccines**. These vaccines are made up of weak viruses. When this is injected into a body the genome starts mutating and reaches a point the virus is unable to cause disease. This method replicates a natural infection, causing the production of T and B cells immune responses.
- **Inactivated vaccines**. These are inactivated viruses using formaldehyde (also known as formalin or formol, a chemical that is commonly used to preserve organs) or heat.

[4] Every nation has their guideline, however, this list has to be taken as baseline guidance.
[5] These concepts are out of the scope of this project, however, were introduced for future reference.

These are classified as noninfectious, stable, and safer compared to the *Live attenuated vaccines*. However, these show low immune response, thus they required multiple doses.

- **Protein-based vaccines**.
  - **Protein sub-unit**. These are antigenic components generated in vitro (i.e., taking place in a test tube). These do not contain any live component from the virus, therefore they are considered safe. However, like *inactivated vaccines*, they show low immune response and need multiple doses.
  - **Virus-like particles**. This type of vaccine contains empty viruses (i.e., no genetic material). They are considered safe and produce strong immune responses.
- **Nucleic acid vaccines**.
  - **DNA vaccines**. They are made up by "*introducing DNA encoding the antigen from the pathogen into a plasmid*" (where plasmid refers to a genetic structure that can be replicated independently). Consider to be safe, however, these types of vaccines are unproven in practice and may cause adverse events when used alone.
  - **RNA vaccines**. They are made up of lipid-coated mRNA of the virus protein. Consider to be safe, however, these types of vaccines are unproven in practice and may cause adverse events.
- **Viral vector vaccines**. The technology used to develop these types of vaccines is *Recombinant DNA*. "*The DNA encoding an antigen from the pathogen is inserted into the virus vector*". Virus vectors refer to "tools" used to deliver genetic material (in simple words, the "instructions" from the virus). There are two types of Viral vector vaccines: Replicating and non-replicating. Both are considered to be safe.

Creating a new vaccine is arduous work that could take more than 10 years and might cost between £200 and $500 million. There are five sequential stages in a vaccine development process (wellcome 2021):

- **Discovery research**. This stage involves laboratory research to induce an immune response. Takes between 2 and 5 years to be completed.
- **The pre-clinical stage**. Test in animals to assess safety. This might take up to 2 years to be completed.
- **Clinical development**. Test in humans, that involves three phases:
  - phase I. Testing for safety. This phase might take 2 years to be completed.
  - phase II. Immune response Understanding. This phase might take 2 to 3 years to be completed.
  - phase III. Assessing whether vaccine protects against decease. This phase might take 5 years to be completed.

- **Regulatory approval**. Submit information on the vaccine to regulatory authorities for review. This can take 2 years to be completed.
- **Manufacturing and delivery**. This requires special facilities highly regulated.

However, concerns about the new virus have making pressure on society resulting in the redesign of the standard vaccine development approach: new collaborative approaches, funding for multiple vaccines, and creation of additional manufacturing and distribution. Pharmaceutical businesses around the world started programs to accelerate a vaccine. Shortly, Moderna and *Pfizer* were the leaders in the race of the development of the COVID-19 vaccine in March 2020. *Johnson&Johnson* and *AstraZeneca* pharmaceuticals joined this race days later (BioSpace 2020). Nowadays, there are 285[6] candidate vaccines (World Health Organization 2021). According to Aurélia Nguyen, managing director of the *COVID-19 Vaccine Global Access Facility*[7], the reason to have many types of vaccines under development "*secured the distribution across all range of population and settings and [...] the huge demand of the vaccines around the world*" (Global Citizen - Gavi The Vaccine Alliance 2020, WHO/N.K. Acquah 2021).

The Food and Drug Administration (FDA) is a federal agency of the Department of Health and Human Services in the USA. Its mission is to protect people's health by ensuring "*the safety, efficacy, and security of human and veterinary drugs, biological products, and medical devices*" (U.S. Food and Drug Administration 2018). The Emergency Use Authorization (EUA) is a procedure to facilities the availability and use of medical products (including vaccines) during public health emergencies. In Europe, the body responsible for authorizing the Covid-19 vaccines to reach the public is the European Commission, after a proper evaluation by the European Medicines Agency (EMA) and consultation with the EU Member States.

The European Commission has been under negotiations to create a portfolio of vaccines for EU citizens. So far, the European Commission has set up contracts with 6 pharmaceuticals where 4 has given a *conditional marketing authorization* and 2 are still under assessment by EMA. A conditional marketing authorization refers to "*an approval of a medicine that addresses unmet medical needs*" (European Medicines Agency 2021). An unmet medical need refers to "*a condition for which there exists no satisfactory method of diagnosis, prevention, treatment, or, even if such a method exists, in relation to which the medicinal product concerned will be of a major therapeutic advantage to those affected*" (Stokx 2019). Table 2.2 shown a list of vaccines[8] that have been approved or are under development[9] in Europe (European Commission 2021). Figure 2.2[10] shows the number of doses secured per vaccine.

[6] Dated: 11-06-2021.
[7] Initiative working for global equitable access to Covid-19 vaccine.
[8] Dated: 03-06-21
[9] A EUA assessment stage.
[10] Source: European Commission. Created on Power BI, © 2021 Microsoft. Dated: 3-Jun-21

Manufacture	Vaccine name	Type of vaccine	No. of doses	Status
Pfizer and BioN-Tech	Cominaty ©	mRNA	2 doses	Approved
Moderna, Kasier Permanent Washington Health Research Institute	COVID-19 Vaccine Moderna ©	mRNA	2 doses	Approved
University of Oxford/AstraZeneca	Vaxzevria ©	Viral vector vaccines	2 doses	Approved
Janssen Biotech Inc., a Janssen Pharmaceutical Company of Johnson & Johnson	COVID-19 Vaccine Janssen ©	Viral vector vaccines	1 doses	Approved
Sanofi Pasteur, GlaxoSmithKline plc (GSK)	-	Protein-based	2 doses	Development ongoing
CureVac N.V. and the Coalition for Epidemic Preparedness Innovations (CEPI)	-	Protein-based	2 doses	Development ongoing

Table 2.2: Covid-19 vaccines approved and ongoing development in Europe.

On the 9th December 2020, the 90-year-old woman, Margaret Keenan, has become the first person to receive a Covid-19 vaccine in the Western world by the bio-pharmaceutical Pfizer/BioN-Tech in the United Kingdom (Sky News 2020).

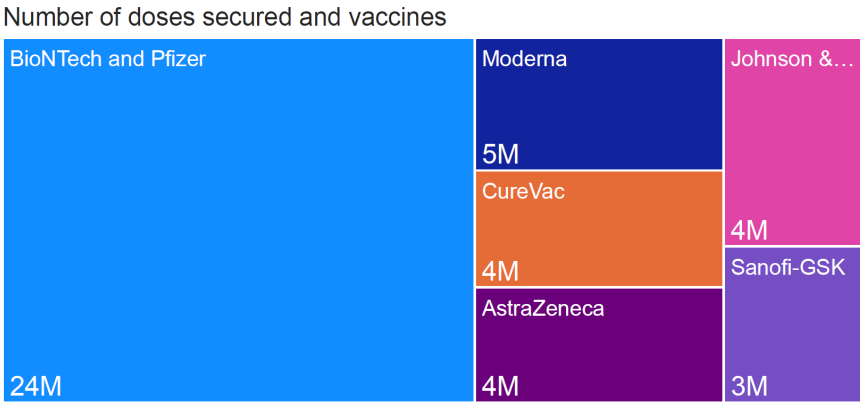


Figure 2.2: Doses secured per vaccine in Europe

2.4 Situation in Ireland

The first Covid-19 case confirmed case in Europe was located in France on the 24th January 2020, an individual who traveled from China (European Centre for Disease Prevention and Control 2021). A month later, the first case was reported on the island of Ireland on the 26th February 2020, a woman whom had a history of travel from the north of Italy (Perumal et al. 2020). A few days later a man located in the eastern side of the country was confirmed to be the first Covid-19 case in Ireland (BBC News 2020). In just one month, all counties had confirmed Covid-19 cases (Cullen 2020). As in many countries around the world, Ireland has been affected by the pandemic. The government quickly took action to diminish the spread of the virus within the country. Some of these actions are enlisted below (RTE 2020*b*):

- Shut all educational institutions and childcare facilities. All education has been taking place online.
- All types of gathering events were called off.
- All businesses were shut, excluding such shops, consider being essentials. Some companies move to *work remotely* mode.
- A period of *lockdown* started, and just essential travel was allowed such as: traveling for essential shopping; from home to work (just in the case the work is essential, health and social care).
- Exercises outside were allowed within 2 km away home.
- No family gathering was allowed, except to provide care to a vulnerable member.

However, Ireland has not been affected just in a social aspect but also economically. A deep recession and massive unemployment has been set in the country as the massive shut of businesses and the strict lockdown period (Chance et al. n.d.) (Burke-Kennedy 2021). Figure 2.3[11] show the Covid-19 adjusted unemployment rate per month, showing the effects of the restrictions and the lockdown.

Figure 2.3: Monthly Covid-19 adjusted unemployment rate.

[11] Source: Central Statistics Office. Created on Power BI, © 2021 Microsoft. Dated: April-21

Later on, the Irish government and the Health Service Executive (HSE) launched a free app called *COVID Tracker* on the 6th of July 2020 (Health Service Executive 2020). This mobile app helps for reporting and tracing as quickly as possible for first onset Covid-19 symptoms. Additionally, this app includes a *Close contact alert* to alarm users (anonymously) when they are close to a person who has been tested positive from Covid-19. Additionally, this app assists to track any symptom shown by advising on what to do (Health Service Executive, Government of Ireland 2020). Figure 2.4 show the user interface of this app.

Figure 2.4: COVID Tracker app

The surveillance of the Covid-19 cases has been carried out by the Health Protection Surveillance Centre (HPSC). The HPSC is an Irish agency specialized in the surveillance of communicable diseases. This is part of the HSE, and in partnership with health service providers and similar organizations, it provides information on diseases (Health Protection Surveillance Centre 2020). HPSC provides daily and detailed reports on confirmed Covid-19 cases in Ireland, including details on each county and other relevant figures. All this information is available on *Ireland's Open Data Portal* and *Ireland's Covid-19 Data Hub*[12] (gov.ie 2020).

"Pandemic wave" has not a strict or fixed definition in the medical or research field. Nowadays, this term is constantly used to describe a period in which the number of Covid-19 cases has been increasing. A "wave" starts with the rising of infected individuals (defined peak), and then a reduction of these cases. The word "wave" implies a pattern of peaks and valleys; during the valley period, it is possible that new cases are "harvested" for the next wave (Wagner 2020). Most of the countries including Ireland have experience three Covid-19 waves so far. Figure 2.5[13] shows graphically these waves. By looking at the graph, it is noticeable that: the first wave happened between March and August 2020; the second wave from August to December 2020; and, the last wave recorded from December to May 2021, the worst recorded.

[12]Due to current "disturbance" on HSE IT, some indicator were paused.
[13]Source: Our World in Data. Created on Power BI, © 2021 Microsoft. Dated: April-21

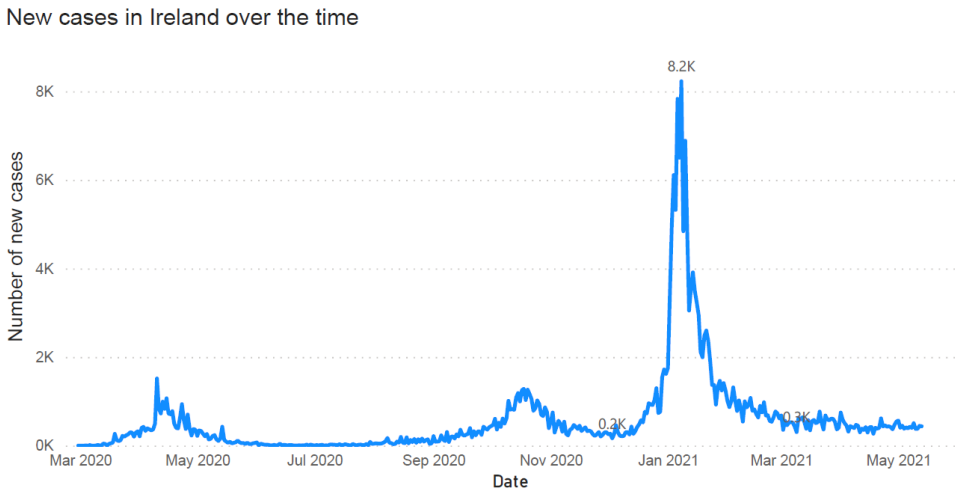


Figure 2.5: Confirmed cases in Ireland over the time

On the 29th April 2021, the Irish government announced a plan for easing restrictions from May to June 2021. Ease restrictions include the re-opening of museums, galleries, libraries, hairdressers, and retail stores, hospitality services reopen, the number of people in in-door/outdoor gathering increased (Dwyer 2021) (gov.ie 2021). However, WHO specialists have warned Irish authorities for a possible *fourth wave* due to possible large gatherings taking place during the ease of restrictions and summer (Digital Desk Staff 2021).

On 26th of December 2020, the first batch of 10,000 doses of Covid-19 vaccines from the bio-pharmaceutical Pfizer arrived in Ireland, and the Covid-19 vaccine roll-out officially started on the 29th of December 2020. The starting point took place at Beaumont and St James' hospitals in Dublin, and university hospitals in Cork and Galway(Curran 2020). The same day, Annie Lynch became the first person in received the Covid-19 vaccine in Ireland at St James's Hospital in Dublin 8 (HSE Press 2020). Currently, the Covid-19 vaccines approved in Ireland are enlisted below:

- Vaxzevria vaccine (from AstraZeneca).
- Moderna vaccine.
- Comirnaty vaccine (from Pfizer/BioNTech).
- Janssen vaccine.

The roll-out is limited by the number of doses available. The priority is to firstly vaccinate all vulnerable individuals and healthcare workers. The vaccination program is divided into groups according to age and high level of risk (this refers to people with other diseases, more vulnerable to the Coronavirus disease 2019) (Health Service Executive 2021). The official groups are enlisted below.

- Group 1: 65-year-old people and older living in care facilities.
- Group 2: healthcare workers such as doctors and nurses.

- Group 3: 70-yea-old people and older
- Group 4: vulnerable people aged 16 to 69 years old.
- Group 5 and 6: people aged 65 to 69, including vulnerable individuals.
- Group 7: people aged 16 to 64, including vulnerable individuals.
- Pregnant women

Figure 2.6[14] shows the total number of vaccines already applied per pharmaceutical nationwide in Ireland until 11th of May 2021[15].

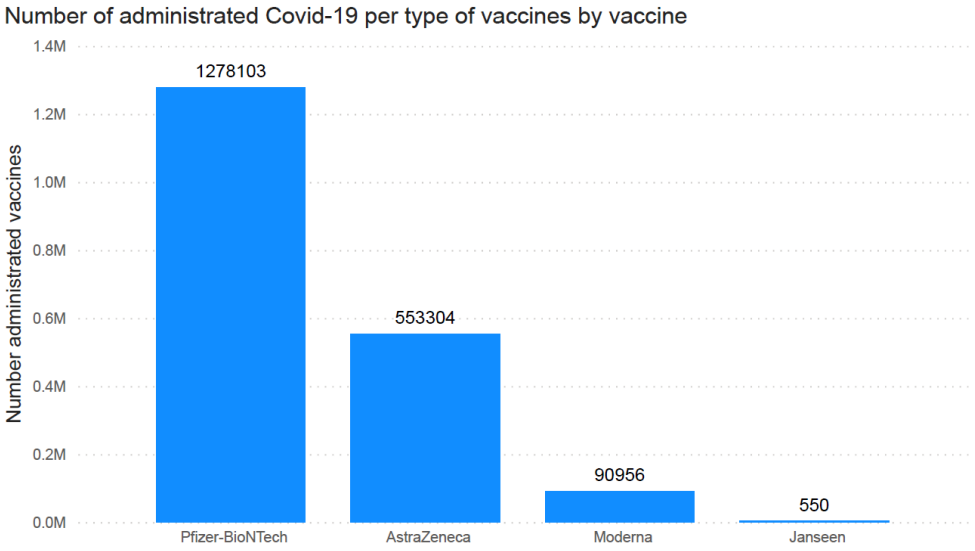


Figure 2.6: Total number of vaccines administered in Ireland.

2.5 Twitter API

Twitter is an American micro-blogging social media on which, as a registered user, it is possible to post, reply, give a like, share and receive short messages called *tweets* (unregistered users are just able to read content). The term *micro-blogging* refers to the activity to frequently post short messages on social media, such as personal activities (Merriam-Webster 2021*a*). Twitter has their headquarters in San Francisco, California, USA and has set up more than 25 offices worldwide (Wikipedia The Free Encyclopedia 2021). Figure 2.7 it is shown the business logo[16].

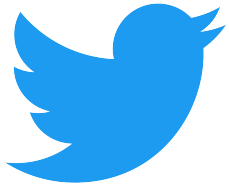


Figure 2.7: Twitter Logotype.

[14] Source: Ireland's COVID19 Data Hub. Created on Power BI, © 2021 Microsoft. Dated: April-21
[15] Due to current "disturbance" on HSE IT, some indicator were paused.
[16] Image taken from Twitter website. Available from: https://about.twitter.com/en/who-we-are/brand-toolkit

Twitter provides a *Develop platform* to exploit Twitter's public and global data, providing tools for businesses, researchers, and more. The products available are shown in table 2.3 (Twitter, Inc. 2020*b*).

Product Name	Description
Twitter API	A set of programmatic endpoints (i.e., touch-points of this communication) to retrieve specific information.
Twitter Ads API	This enables to maintain and track campaign.
Twitter for Websites	Allow to stream in live content into external products.
Labs	This allows developers to try and test previous or new API products for the next generation, sharing feed-backs and suggestions.

Table 2.3: Twitter Developer Platform products.

NOTE: *As this project required access to Twitter API, beyond this point it will be described in more detail the Twitter API product, as other tools were not relevant for the project.*

The **Twitter API** is used to collect and analyze data, as well as engage with conversations on Twitter (Twitter, Inc. 2021*i*). As a way to support developers who have just started using Twitter API, there is well-structured documentation available on *https://developer.twitter.com/en/docs/twitter-api* including Q&A, tutorials, and community forums for questions. To start using this tool, it is required to apply for a *Twitter Developer Account*. Within the application, it is necessary to indicate a use case according to the work intended to carry out to "*protect the people that use Twitter*" (Twitter, Inc. 2021*d*). It is possible to apply to the following product trackers:

- **Standard**. This is the *default* product track used by many developers. This includes learning, teaching, or having fun.
- **Academic Research**. This product track is aimed for academic purposes to give more functionally and access than the Standard product, including access to the *full-archive search endpoint* and enhanced filtering.
- **Business**. For businesses that depend on Twitter's data, giving access to endpoints and custom access. *An application is required to send*.

Twitter has different access tiers depending on the developer's needs. Across tiers, there are two supported version available: *v1.1* and *v2* (this last one is indented to replace all access tiers on v1.1 in the future) (Twitter, Inc. 2021*j*). In table 2.4 shows the list of *access tiers* available where the free version is the *Standard v1.1*. Other tiers charge an amount of money according to the data needed except for *product trackers regarding Academic Research*, providing access to v2 and v1.1. Each access tier has specific *rate limits* for their endpoint. As every day many requests are sent to the Twitter API, the rate limits are used as a strategy to control the volume of daily requests and provide reliability to the developer's community (Twitter, Inc. 2021*e*).

Name	**Version**	**Description**
Early Access	v2	This is the new Twitter API version that would replace version 1.1 in the future, which improves accessibility and experience. New features are: request specific fields and objects; more detailed data available; conversation tracker including new fields; enhances free access for the *Academic Research* product tracker, including access to full-archive search and other v2 endpoints.
Standard	v1.1	Free access tier. It is possible to post content, collect tweets, get tweet timelines, manage user accounts, send events, create welcome messages, get trends, get *geo* information, and more.
Premium	v1.1	This is designed for those who are looking to scale up, accessing to *enterprise* quality features. By default, includes a free sandbox (for try and testing environments) and rate limits.
Enterprise	v1.1	This offers the highest level of access. Beyond the *Premium* access, this is perfect for those who want to scale up their businesses but need more reliable access. Useful to tailor needs. Depending on the contract terms and the account granted is the level of accessibility gained.

Table 2.4: Twitter API. List of available access tiers.

Once the application is approved by *The Twitter Dev team*, it is time to set up the Developer Account by creating the first *App*. Consequently, it is possible to generate a set of credentials to start requesting data from the API (Twitter, Inc. 2021*d*). Table 2.5 shows the name and description of all credentials.

Name	Description
API Key	A set of programmatic endpoints (i.e., touch-points of this communication) to retrieve specific information.
API Key Secret	Username.
Access Token	Password.
Access Token Secret	Represents the Twitter account that owns the App.
Bearer Token	This represents the Twitter account that owns the App. The only difference is that this requires OAuth 2.0 authorization (this method is for only-read access to public information) (Twitter, Inc. 2020*c*).

Table 2.5: Twitter API. List of Credentials.

After getting ready credentials, now it is possible to call an endpoint via Hypertext Transfer Protocol (HTTP) request. Depending on the access tier and version selected, the route path for an endpoint could be (Twitter, Inc. 2021*f*)(Twitter, Inc. 2021*k*):

- For **version 1.1**, the route path is: https://api.twitter.com/1.1/{object_name}/{endpoint_name}. Although, it is suggested to migrate to the version 2 by following the *migration material* available on the Twitter API documentation.
- For **version 2**, the route path is: https://api.twitter.com/2/{object_name}/{endpoint_name}.

There are different approaches to call an endpoint:

- Make a request from a window terminal using the command *curl* and sending the *Bearer Token* in the *header*. The response received from this call approach is in JSON format.

```
curl --request GET 'https://api.twitter.com/2/tweets/search/recent?
    query=from:twitterdev' --header 'Authorization: Bearer
    $YOUR_BEARER_TOKEN'
```

- Using a specific language program such as Python, JavaScript, Java, and Ruby. In Python, it is needed to install the library `request` by running `pip install request` in the terminal. A set of example codes are available on GitHub in https://github.com/twitterdev/Twitter-API-v2-sample-code. An example is attached to this report in Appendix A.1. The response received from this call approach is in JSON format.
- It is possible to use tools, libraries, or clients as long as the endpoint (and version) is supported. The official tools developed, supported, and maintained by Twitter are shown in Table 2.6. Additionally, Twitter has enlisted in their documentation

community-supported libraries on a different programming language such as .NET, C++, Java, R, and so on. However, most of these tools have not been tested by the Twitter team. The whole list is available from *https://developer.twitter.com/en/docs/twitter-api/tools-and-libraries*.

Language	Description
JavaScript/Node.js	`Autohook`, a tool to configure, manage and receive account events such as receiving tweets or direct messages in real-time with one connection (Twitter, Inc. 2021*b*)).
Python	`serach-tweets-python` client. This support the Recent Search and Full-Archive Search endpoints. The response received from this call approach is in JSON format.
Ruby	`serach-tweets-ruby` client. This support the Recent Search and Full-Archive Search endpoints. The response received from this call approach is in JSON format.

Table 2.6: Twitter API. Official tools for requests.

- Using Postman, a desktop and web application to send HTTP requests from a graphical user interface. Figure 2.8 shows the web application platform.

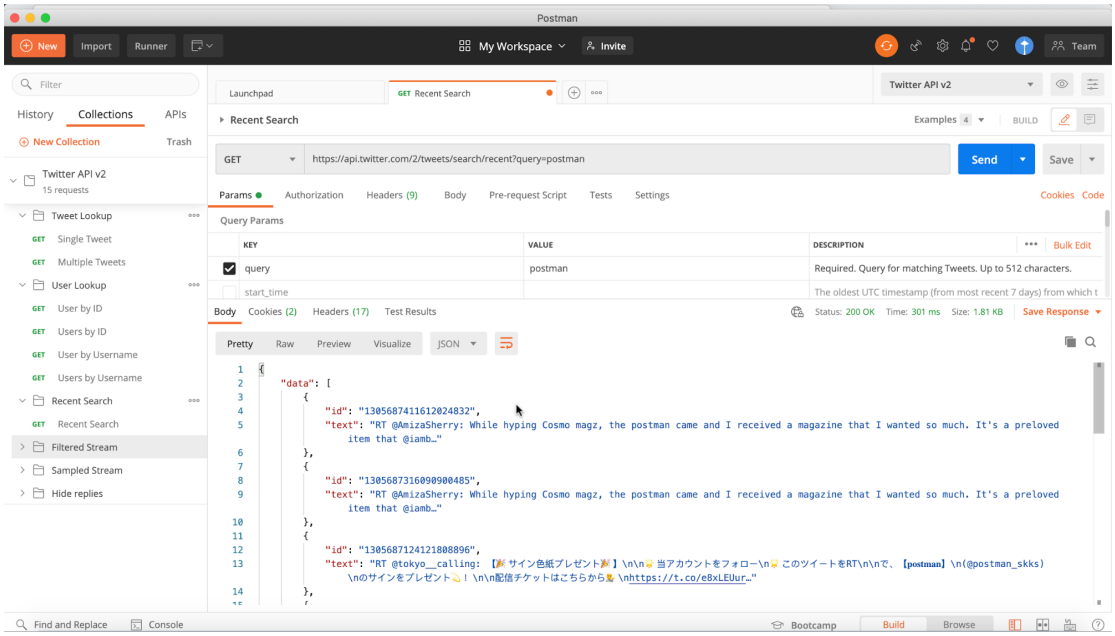


Figure 2.8: Twitter API. Postman web application user interface.

The Twitter API provides access to a variety of objects available, that are enlisted as follows (the objects *Tweet*, *User* and *Place* will be described in more detail as these were more relevant for the project, other objects will be described briefly) (Twitter, Inc. 2021*a*):

- **Tweets**. This object represents the basic structure of all things in Twitter. This is the parent of other objects like *user*, media, poll and *place*. It is possible to request additional fields by indicating the parameter `?expansions=referenced\_tweets.id`.

Table 2.7 shows a description of the default and optional fields that belong to this object.

- **Users**. This object contains user's metadata. The Tweet object may also contain the object User, which is included indicating `?expansions=author_id` or `?expansions=in_reply_to_user_` for default fields and indicate parameter `user.fields` to incorporate additional user fields. Table 2.8 shows a description of the default and optional fields that belong to this object.

- **Direct Messages**. This object could be used to send events or receive Direct Messages (useful to create Direct Message bots).

- **Trends**. This object contains information regarding the place and the most popular topics based on hashtags.

- **Media**. This represents an object such as a picture, video, or animated GIF. These are used for many endpoints, and those might be included in other objects such as Tweet object, Direct Message object, and User Object. The size restrictions for each type of Media object are: for images 5MB; for animated GIF 15MB; and, for video 15MB.

- **Places**. This information comes from users who tagged their location on the tweet. This may or not may exist, and if those could be included in the Tweet object. It contains information from the user's place such as country, city, latitude and longitude, and so on. This is available for expansion including `?expansions=geo.place_id` on the route path, and including more fields including the parameter `place.fields`. Table 2.9 shows a description of the default and optional fields that belong to this object.

- **Entities**. This object provides additional information regarding the content of a tweet. This includes details about hashtags, URLs, user mentions, and symbols.

Field name	Type	Description
id (default)	string	Unique identifier of the tweet.
text (default)	string	Text of the tweet. It is in UTF-8 format.
attachments	object	Indicates type of media attached to tweet.
author_id	string	Unique identifier of the user who post the tweet.
context_annotations	array	Contains context annotations regarding the tweet.
conversation_id	string	New field to enhance tracking of conversations related to a tweet. This includes direct replies , replies of replies.
created_at	data (ISO 8601)	Date and time tweet was created.
entities	object	Provide more information about hashtags, URLs, user mentions, and cashtags (like hashtags but using the sign $).
geo	object	Contain information about the tagged location of the user (if exists).
in_reply_to_user_id	string	This is the unique identifier that could be used to determine if the tweet is a reply to another tweet.
lang	string	Specify the language of the tweet.
non_public_metrics	object	Represent engagement metrics non seen by the public such as impression_count, url_profile_clicks and user_profile_clicks.
organic_metric	object	Engagement metrics tracked in an organic way. This could include metrics such as like_count, reply_count, retweet_count, and url_link_clicks.
possibly_sensitive	boolean	Based on the link contained on the tweet, this indicates if the URL may or may not contain sensitive content.
promoted_metrics	object	This field is similar to the organic_metric, the only difference is that these are metrics when a tweet is promoted through Twitter Ad.
public_metric	object	Public engagement metrics for the tweet. Similar to organic and non-public metric fields, however this object just return metrics that can be seen by any user.
referenced_tweets	object	A list of tweets related to the tweet. For instance, if the parent of the tweet is a retweet, this would be included in the list.
reply_setting	string	Shows who can reply to the tweet. Options are: everyone, mentioned_users and followers.
source	string	The name of the application used to post the tweet.
withheld	object	Contains withholding details (if exists) in the tweet.

Table 2.7: Twitter API. Tweet object fields.

Field name	Type	Description
id (default)	string	Unique identifier of the user.
name (default)	string	Name of the user defined on the profile settings.
username (default)	string	Screen name that is used to identify the user. This is unique but user can modified.
created_at	date (ISO 8601)	Date and time when the user was created on Twitter.
description	string	Description provided by the user and it is shown on the profile.
entities	object	Provides more information about hashtags, URLs, user mentions and cashtags contained in the user's description.
location	string	The location indicated on the user's profile (if exists).
pinned_tweet_id	string	Unique identifier of this user's pinned Tweet.
profile_image_url	string	Provides the URL to the profile user image.
protected	boolean	Indicates if the user has chosen to protect their tweets (private).
public_metrics	object	This filed contains details on the activities of the user.
url	string	The URL specify in the profile of the user (if exists).
verified	boolean	Indicates if the user is a verified Twitter User.
withheld	object	Contains withholding details (if exists) in the user.

Table 2.8: Twitter API. User object fields.

Field name	Type	Description
full_name (default)	string	Name of the place in a longer-form. This may include city, country, and other fields contained from the object Place.
id (default)	string	Unique identifier of the place.
contained_within	array	Return a list of identifiers contained in a referent place.
country	string	Name of the country.
country_code	string	Country code the place belongs to.
geo	object	This field contains details regarding the place in GEOJSON format.
name	string	The short name of the place.
place_type	string	Specify the particular type of information by the place such as city, or country.

Table 2.9: Twitter API. Place object fields.

Twitter API has a huge variety of endpoints available (depending on product tracker and version). Related to tweets, the endpoints available for version 2 are:

- **Tweet lookup**. These endpoints return a Tweet object that describes a specific tweet, including text, created at date and URL. The route path to access the endpoint is `https://api.twitter.com/2/tweets?ids=`. More fields could be included by adding `&expansions=author_id` and `&user.fields=` (Twitter, Inc. 2021*h*).
- **Recent search quick start**. This endpoint allows collecting public tweets from the last seven days. It is possible to filter the content by building a query. The route path to access the endpoint is `https://api.twitter.com/2/tweets/search/recent?query=`. It is possible to expand the response by including `&expansions=` (Twitter, Inc. 2021*g*).
- **Full-archive search**. This endpoint is just available for *Academic Research* product tracker. This allows collecting tweets dated from March 2006 (back to the first tweet posted.) This endpoint sends up to 500 tweets per request. The route path to access the endpoint is `https://api.twitter.com/2/tweets/search/all?query=`. It is possible to expand the response by including `&expansions=` (Twitter, Inc. 2021*g*).
- Other endpoints (not described in more detail as these are not relevant for the project) (Twitter, Inc. 2021*i*):
  - **Timeline**. This endpoint provides access to tweets published by a specific Twitter account.
  - **Filtered stream**. This endpoint enables to filter in a real-time stream of public Tweets.
  - **Sampled stream**.
  - **Likes**. These endpoints allow collecting tweets that have been liked, give likes to tweets, or unlike tweets.

When calling an endpoint to search tweets, it is needed to build a query to filter the data according to the developer's needs. Depending on which product track you have acquired, the query could be up to 512 characters long for *Standard* product track and 1024 characters long for an *Academic Research* product track. A query is made up of different types of operators that could represent a keyword, a place, or hashtags. The table 2.10 show a list of the most relevant operators [17]. There are two types of operator, explained as follows (Twitter Inc. 2021*c*):

- **Standalone operators**. These could be keywords or hashtags that could be used alone or together.
- **Conjunction-required**. General filters that cannot be used by themselves as they could return an enormous answer. These have to be used with at least one standalone operator. Generally, these have the form `operator:value`.

[17] The whole list is available from *https://developer.twitter.com/en/docs/twitter-api/tweets/search/integrate/build-a-query*

Operator	Type	Description
`keyword`	Standalone	This is a keyword to match within the tweet.
`emoji`	Standalone	Like keywords, tweets could be also filtered using emojis.
`"exact phrase"`	Standalone	This is used when filtering tweets by using matching words.
`#`	Standalone	To filter hashtags within the tweet.
`@`	Standalone	This operator is used to find out tweets that users have been mentioned within the body of the tweet.
`from:`	Standalone	This operator matches tweets from a specific user.
`to:`	Standalone	This matches any tweet that has been replied by an specify user.
`conversation_id:`	Standalone	This operator allows to filter tweets that belong to a particular conversation.
`place:`	Standalone	Matches all tweets that have been tagged in a specific city (if exists).
`place_country:`	Standalone	Matches all tweets that have been tagged in a specific country (if exists).
`is:tweet`	Conjunction-required	This operator is used to select just tweets. This is useful in its negative form which is `-is:tweet` to return replies or quotes as an example.
`is:reply`	Conjunction-required	This operator is used to select just replies.
`is:quote`	Conjunction-required	This returns tweets that have been marked as quotes, known as tweets with comments.
`is:verified`	Conjunction-required	This returns tweets that had been posted by verified users (blue mark next to the username0.
`lang:`	Conjunction-required	Returns all tweets form a specific language.

Table 2.10: Twitter API. Operators to build a query.

To group Standalone and Conjunction-required operators, it is possible to use boolean operators, which are shown in table 2.11. The boolean operator `AND` is always performed first, therefore to avoid uncertainty it is suggested to use the group operator.

Boolean operator	**Description**
AND	This is represented as a space between operators. An example would be `snowday #NoSchool`, which means the query would filter all tweets that contain the keywords snowday **and** the hashtag NoSchool.
OR	This is represented with the word `OR` between operators. An example would be `cat OR dog`, which means the query would filter all tweets that contain the keywords cat **or** dog.
NOT	.This is useful to negate operators. The way to specify the NOT operator within a query is with a dash(-). An example could be `cat #meme -grumpy`, which means filter all tweets with the keyword cat and hashtag meme and that not contain the word grumpy.
Grouping	It is possible to group operators by using parenthesis. For example `(grumpy cat) OR (#meme has:image)`, that means we are filtering the tweets by two groups: two keywords grumpy and cat; or the hashtag meme and tweets that contain an image.

Table 2.11: Twitter API. Boolean Operators.

2.6 Sentiment Analysis

Data could be seen as the "new oil" because of its powerful applications after "its refining processes" to transform it into knowledge. Despite the discussions towards ethical issues, the use of the data has brought many helpful and useful applications to the society (Forbes 2019). This all is possible with **Machine Learning** (ML) that consists of algorithms that utilize data to learn, improving predictions and decision-making over time (IBM 2020*a*). Different learning scenarios depend on the data available for the training stage. The most common types of learning are (Mohri et al. 2018):

- **Supervised learning**. In this type of learning, the algorithm receives a set of labeled examples in the training stage to make predictions on unseen data. For instance, a set of emails with labels spam/no-spam for classification problems.
- **Unsupervised learning**. This receives a set of data unlabelled. This scenario is difficult to evaluate as the lack of labeled examples. Examples of this problem are clustering and dimensional reduction.

- **Semi-supervised learning**. In this scenario, the learner receives a set of labeled and non-labeled examples.
- **Reinforcement learning**. This is when training and test stages are "intermixed" where the algorithm interacts with the environment and receives an immediate reward for each action.

There are a myriad number of problems ML can deal with. The most successful applications are those that improve the decision-making process, improving many companies and institutions that take advantage of the power of ML. For instance, some of these practical applications applied in the real world are (Muller & Guido 2017):

- Detect unusual activity in credit card transactions.
- Identify topics in a set of text or documents.
- Detect if an email is a spam or not spam.
- Classify flowers by type.
- Group customers according to similar preferences.

Natural Processing Language (NPL) is an interdisciplinary field that combines linguistics, computer science, and artificial intelligence. This allows machines to interpret, understand and produce human languages. All applications are often built upon machine learning models and a set of texts/conversations generated by humans to humans. Some examples of these applications are: translating from one language to another such as Google Translate; digital assistance such as Alexa from Amazon; and, speech-to-text dictation (Metwalli 2020)(IBM 2020*b*).

Sentiment Analysis (also known as Opinion Mining) is a NPL technique used to classify a text into a specific opinion (polarity like positive, negative, or neutral). Additionally, there are Sentiment Analysis models that are build to interpret emotions (joy, anger, surprise), urgency feeling (no urgent or urgent), and intentions (not interested or interested) within a text (Monkey Learn n.d.). This has become recently popular with the massive use of social media and online reviews. People share their thoughts and feelings about different topics and products through the internet, more precisely on social media platforms such as Facebook and Twitter, or e-retail websites like Amazon, or their opinions about other things like reviews about apps on Google Play or reviews about movies on IMDb website. Big companies have taken advantage of this data available to analyze the opinion of the public towards a specific product or topic and enhance the making-decision to improve the service or the product. Examples of its applications are: interpretation of reviews from a specific product, understand customers feeling about specific topics, customer service, brand monitoring, market research, and so on. Figure 2.9 shows a graphical example of text classification, where these comments may express their feedback about a service in a hotel, airline, restaurant, and so on.

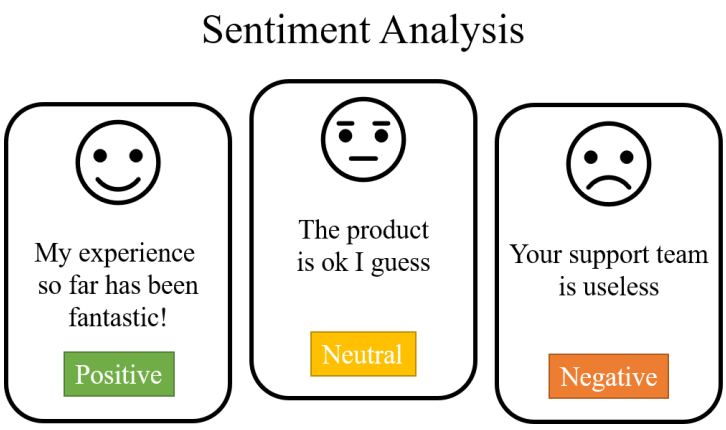


Figure 2.9: Example of Sentiment Analysis

2.7 Limitations on Sentiment Analysis

Sentiment Analysis has some challenges as the complexity of the grammar and language in itself and the constant evolution of the language (not just English but many other languages around the world). Especially in social media, where people can express their opinion in different ways using slang and expressions that machines do not understand. The challenging problems that might affect the accuracy of the SVM algorithm in the modeling stage are (Eremyan 2020):

- **Irony**. As defined by Merriam-Webster, irony refers to the use of words to communicate the opposite of the meaning (Merriam-Webster 2021*b*). Examples of irony could be "A fire station burns down", or "A marriage counselor files for divorce".
- **Sarcasm**. This is the use of words ironically to insult/irritate someone. For instance, imagine the situation in which someone says something very obvious, and the other says "Really, Sherlock? No! You are clever". This phrase is cataloged as a sarcastic remark. There are different types of sarcasm: Propositional, Embedded, Like-prefixed, and Illocutionary.
- **Ways of negations**. There are different ways sentences people can negate and what words are affected by this negation (a range is present). There are several ways to express negation such as using prefixes or suffixes, also negations could be implicit such as "This will be the first and last time I visit your friend", or using explicit words for negation such as "not".
- **Multi-polarity**. Sometimes a document or opinion could show multiple opinions. Generally, this happens for instance when a user is given feedback about a product this is describing positive opinions about some features and bad options to other features.
- There is no approach to deal with dependent words.

2.8 Survey on sentiment classification techniques

There are several ML algorithms in the Sentiment Analysis domain and have been discussed in various academic sources. The techniques considered for the modeling stage were Naive Bayes (NB), Support Vector Machine (SVM), Maximum Entropy (ME), and Neural Network (NN) since these algorithms were commonly tested on academic work. Finally, the algorithm chosen was SVM because of its outperformance. This is discussed in more detail in the following sections.

2.8.1 Naive Bayes (NB)

This is a supervised ML method that is based on the *Bayes' Theorem* for clustering and classification problems (Dey 2016). Bayes' Theorem is also known as Bayes' Rule, used as an alternative to computing the conditional probability (Pollard 1997). However, in this case, this will not be employed for Sentiment Analysis as the dataset will contain a large number of features after vectorization[18]. Therefore, NB enhances the Bayes' Theorem to handle a large dimensional dataset, assuming each data point is independent of each other (Paruchuri 2015). The formula 2.1 shows the conditional probability.

$$P(X|y_i) = \prod_{i=1}^{m} P(x_i|y_i) \tag{2.1}$$

where:

X is the feature vector $X = \{x_1, x_2, ...x_i\}$, and x_i is a word i.

y_i is class label (e.g., i could be positive, negative, or neutral).

There are different approaches that according to the distribution of the data (Ray 2017):

- **Gaussian Naive Bayes**. Assumes that features follow a normal distribution. This is also known as Naive Bayes.
- **Multinomial Naive Bayes (MNB)**. This approach is better for discrete values (like word counts in text) thus this might show a better performance than NB for text classification.
- **Bernoulli Naive Bayes**. Assumes there is a binomial distribution, e.g., when features have valued one or zero.

This ML algorithm is fast comparing to other methods. It is used commonly for Text Analysis and Sentiment Analysis as its power to handle high dimensional datasets, able to create fast predictive models (Mandloi & Patel 2020). Nevertheless, the downside of the NB approach is that this does not evaluate the relation among features such as part of speech tags and negation (Neethu & Rajasree 2013).

[18] In simple words, convert text into number.

2.8.2 Support Vector Machine (SVM)

Support Vector Machine is another supervised ML method that is useful for classification problems, regression analysis, and outliers detection. The goal of this algorithm is to classify data into different groups by drawing a hyper-plane between them. In other words, drawing a margin between classes in a way that the distance between the margin and the classes is the maximum, minimizing the error of classification. This is illustrated in figure 2.10 in which the image in the left represents all possible hyper-planes that could be selected, and in the right one the optimal hyper-plane that maximizes the distance between classes (i.e., positive and negative). The vectors that defined this decision boundary are called support vectors (Gandhi 2018).

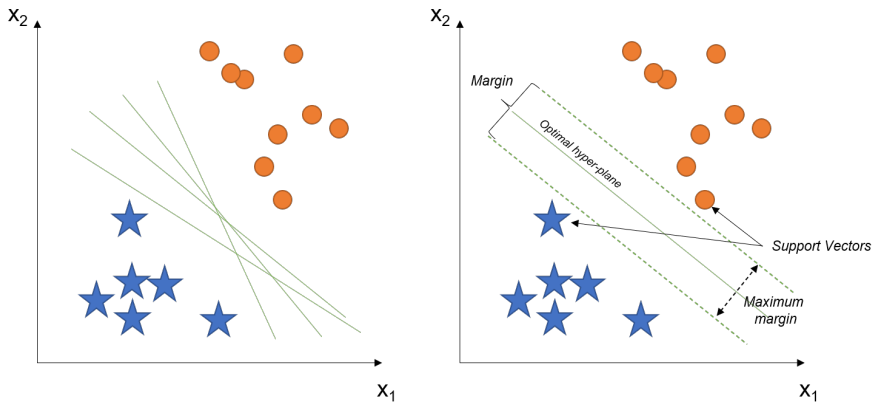


Figure 2.10: Graphical representation of Support Vector Machine in 2 dimensions.

SVM uses the hyper-plan representation shown in formula 2.2 (Neethu & Rajasree 2013).

$$g(X) = w^T \phi(X) + b \tag{2.2}$$

where:

X is the feature vector $X = \{x_1, x_2, ...x_i\}$, and x_i is a word i.

w is the weights vector.

ϕ kernel function.

b is the bias vector.

The kernel function is a mathematical function used to "draw" the optimal hyper-plane by mapping the data into a higher-dimensional space for easier linear and non-linear classification problems. The most popular are: linear, polygonal, radial basis function (RBF), and sigmoid (Ashis 2012). Figure 2.11 illustrates these kernel functions graphically [19].
The same as NB, the advantage of SVM is its power to handle large dimensional spaces. In Sentiment Analysis, as long as the classification problem could be linearly separable, the volume of features in the dataset would not be a problem. This approach is particularly preferred due to the significant accuracy score, less computation power, and effectiveness on

[19] Source: https://scikit-learn.org/stable/modules/svm.html

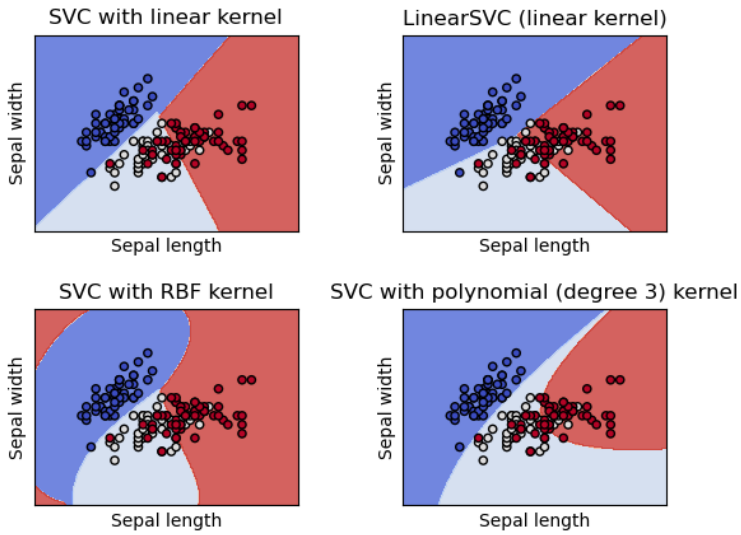


Figure 2.11: Graphical representation of kernel functions.

high-dimensional spaces. However, a disadvantage is the *blackbox* process in its methodology as it is difficult to look into the nature of classification and which words are important (Bhuta et al. 2014).

2.8.3 Maximum Entropy (ME)

It is an alternative technique as its good performance in numerous NPL applications (Berger et al. 2002). This classifier tries to maximize the probability of a document into a class by computing the conditional distribution. This algorithm calculates the probability of any distribution, not making independent assumptions on the features. This is shown in the formula 2.3 (Nigam et al. 1999).

$$P_\lambda(y|X) = \frac{1}{Z(X)} \sum_{i=1}^{n} \lambda_i f_i(X, y) \tag{2.3}$$

where:

X is the feature vector $X = \{x_1, x_2, ...x_i\}$, and x_i is a word i.

y is class label (e.g., positive, negative, or neutral).

$Z(X)$ is the normalization factor to ensure the probability.

λ_i is the weight coefficient.

$f_i(X, y)$ is the feature function which $f_i(X, y) = \begin{cases} 1 & X = x_i \text{ and } y = y_i \\ 0 & \text{otherwise} \end{cases}$.

2.8.4 Neural Network (NN)

The idea of NN is based on the concept of "neurons" in the biology field. The center of the human body system is the brain. The human body produces internal/external stimuli converted by receptors (biological transactors that convert energy into electrical impulses) connected to the central nervous system (Rudge et al. 2020). All this information goes

straight to the brain that contains a neural net. The dendrites receive electrical signals from the "stimuli". The "some" processes these signs and the proteins (output) are carried by the axons to the next neuron's dendrites. A single neuron has three main parts, enlisted below. Figure 2.12[20] shows the structure of a neuron (Woodruff 2019).

- **Axons**. A long thin structure used to communicate to other neurons.
- **Dendrites**. These structures receive signals from axons and determine if any action will be triggered by a neuron.
- **Soma**. Neuron's core. Also known as cell body. This is the part of the neuron where the nucleus is stored. The nucleus is like the "heart" of the neuron that produces proteins to be transported to another neuron.

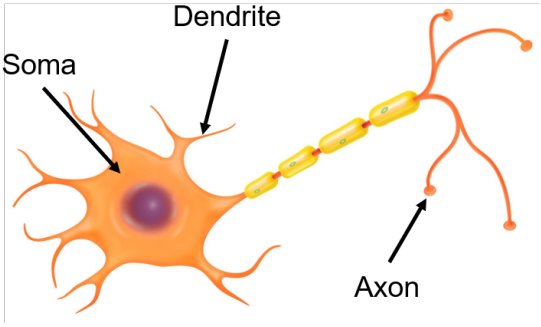


Figure 2.12: Parts of a neuron.

A NN in ML works in the same matter. This works on layers, and mainly includes the following steps: (1) the input layer (like the dendrite) receives the information; (2) the hidden layers (like the body cell) process this information, storing knowledge for the next neuron; (3) finally, the output layer (like dendrites) sends outcomes (Dey 2016). An artificial neuron in NN has the structure shown in figure 2.13.

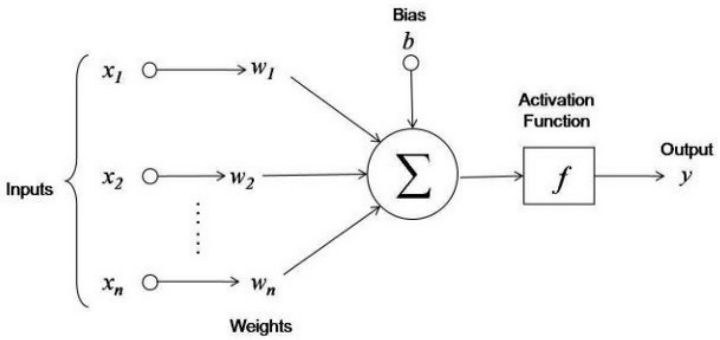


Figure 2.13: Parts of a neuron in Neural Network.

This structure consists of a set of connected neurons called nodes. These are mathematical functions that receive inputs. Each one is weighted, then all of them are summed up and added a bias value. After that, it is passed through an activation function for the next neuron or outcome. An activation function converts this value into another the next neuron "can use". There are different types of activation functions, which are used for non-linear problems: sigmoid function, Tanh function, and Rectified Linear Unit function (Siddiqui 2019).

[20] Image taken from: https://www.dkfindout.com/us/human-body/brain-and-nerves/nerve-cells/

The Recurrent Neural Network (RNN) is a type of NN, a neural sequence that iterates along with sequence elements (Malik 2019). This can deal with text classification as its ability to retain information from what has been processed. This easily allows to "learn" for an extended number of iterations (Du et al. 2021).

2.8.5 Discussion on the ML algorithms

For sentiment classification on text, most of the ML techniques show excellent performance and have been studied and compared by researchers. Pang and Lee studied the performance of three ML algorithms for sentiment classification. The results of Pang and Lee (2002) are shown in Table 2.12, where every test was running with a different number of features. Despite the fact that NB had a better accuracy average, SVM tended to outperform when looking at each test. However, the difference among accuracy scores was slight. Similar work was done by Go et al. (2009), obtaining similar results as Pang and Lee's work. They found out SVM showed a tendency to perform better than other classifiers, although with slight differences.

	Classification accuracy		
Test	**NB**	**ME**	**SVM**
(1)	**78.7%**	N/A	72.8%
(2)	81%	80.4%	**82.9%**
(3)	80.6%	80.8%	**82.7%**
(4)	77.3%	**77.4%**	77.1%
(5)	81.5%	80.4%	**81.9%**
(6)	77.0%	**77.7%**	75.1%
(7)	80.3%	81.0%	**81.4%**
(8)	81.0%	80.1%	**81.6**%
Average	**79.7**%	69.7%	79.4&

Table 2.12: Pang and Lee's results.

The research work from Neeth and Rajasree (2013) showed that in spite of a better precision score for the NB algorithm, SVM had higher recall and accuracy scores. They obtained an accuracy of 90% for SVM, whereas NB was 89.5%. Once again, there was a slight difference between these accuracy scores. Maharani (2013) performed an examination on the following ML algorithms for sentiment analysis: SVM, ME, MNB, and k-NN. In this study, ME and MNB algorithms tended to outperform. However, they found that SVM displayed a better accuracy scored on average, although differences were not large. The results are shown in table 2.13 (Maharani 2013).

Method	**Accuracy**					**Average**
SVM	84.0%	78.4%	79.6%	**82.6%**	82.5%	**81.4%**
ME	**84.2%**	80.7%	**83.6**%	80.5%	76.9%	81.2%
MNB	83.7%	**81.5**%	78.2%	77.9%	**84.5%**	81.2%
k-NN	72.6%	75.2%	72.9%	70.0%	70.1%	72.1%

Table 2.13: Maharani's results.

Related to NN, Ak-Smadi et al. (2018) compared RNN and SVM findig out SVM outperforms on sentiment identification, althought RNN had shown a better execution time. Table 2.14 shows the results.

Methods	Accuracy
CNN	82.7%
RNN	87%
SVM	**95.4%**
Baseline	76.4%

Table 2.14: Ak-Smadi et al. results for sentiment identification.

In almost every case, researchers and practitioners accorded that SVM was better. Therefore, SVM was chosen for the modeling stage in this project.

2.9 Labeling

As it was expected to collect a large number of tweets for the modeling stage, it was impossible to manually categorize thousands of tweets into positive, negative, or neutral opinions. As a result, a survey on labeling approaches was conducted. The options considered for the labeling process were *MonkeyLearn*, *Microsoft Azure*, and the three lexicon-based tools *VADER*, *SentiWordNet*, and *TextBlob*. Finally, the lexicon-based tool *VADER* was chosen as has been, which it is discussed in the following sections.

2.9.1 MonkeyLearn

MonkeyLearn is a ML platform for Text Analysis. This implements tools for Topic Classification, Sentiment Analysis, and Intent Classification[21]. Moreover, this includes a tool for the extraction of specific pieces of data from documents (Maguire 2021). With a friendly interface shown in figure 2.14, it is easy to use for those with no programming knowledge. For more experienced users, an API is available for different programming languages, including Python. This platform provides the opportunity to train a model with a dataset imported by the user.

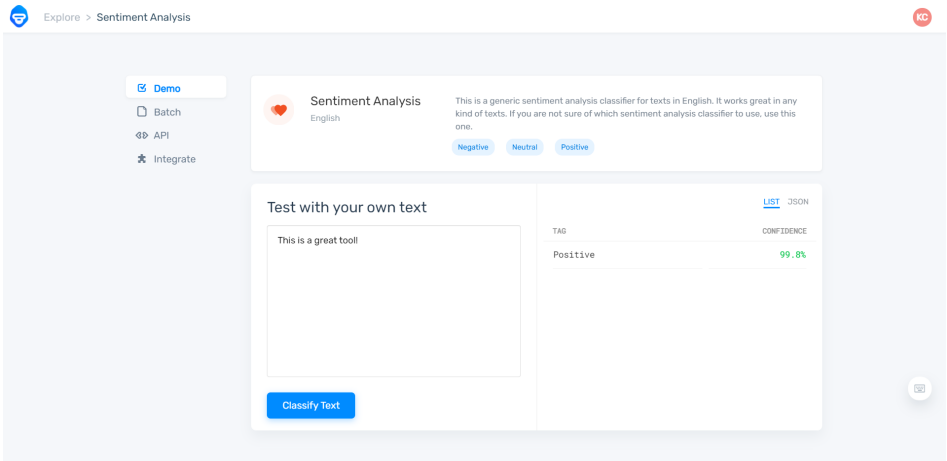


Figure 2.14: User interface from MonkeyLearn platform.

[21] Classify text based on intent, e.g., complaint, request, feedback.

Since the free plan is limited to 300 calls to API, the options to upgrade are *API* and *Studio*, including a monthly fee charge. Fortunately, MonkeyLearn offers an *academic plan* for free, which is similar to the API plan. It is required to contact the MonkeyLearn staff and provide information about your eligibility to get access to the academic plan. MonkeyLearn has well-detailed documentation which explains how to use the API, rate limiting, query limits, example codes, and more. The API is easy to use in Python by installing the library `monkeylearn` in the environment. First off, a model has to be created on the MonkeyLearn dashboard to generate the API Key and the Model ID. With this information, it is possible to use the MonkeyLearn library in Python to call the API and get the sentiment of a text. An example code in Python in Listing 2.1.

```
from monkeylearn import MonkeyLearn

ml = MonkeyLearn('[YOUR_API_KEY]')
data = ['first text', {'text': 'second text', 'external_id': 'ANY_ID'}, '
    ']
model_id = '[MODEL_ID]'
response = ml.classifiers.classify(model_id, data)
print(response.body)
```

Listing 2.1: MonkeyLearn code example in Python.

This platform is a good option for the labeling process of the tweets, although it was discarded due to its ethical considerations stated in the Privacy Policy and Terms of Use. Despite the fact that MonkeyLearn states that "*Your data is considered confidential information by MonkeyLearn as set forth in our Privacy Policy,*" it is not clear whether the data (in this case, tweets) send to the MonkeyLearn API is stored by MonkeyLearn. Due the headquarters are located in San Francisco, CA, U.S., in their Privacy Policy is stipulated that *"If you are located in the* ***European Union*** *[..] with laws governing data collection and use that may differ from U.S. law, please note that you are* ***transferring information, including personal information****, to a country and jurisdiction that does not have the same data protection laws as your jurisdiction, and you consent to the transfer of information to the U.S. and the use and disclosure of information [..]*" (MonkeyLearn Inc. 2019). This is a critical point as it has to be aligned to the terms and conditions of Twitter (primarily, no share collected tweets with third parties).

2.9.2 Microsoft Azure

Azure is a cloud computing platform that provides more than 200 services and products, including storage, analytic, and networks (Microsoft Corporation 2021*e*). Within all these products, Microsoft Azure offers *Text Analytics*, a text-mining AI service using NPL to uncover hidden insights on text, including sentiment analysis (Microsoft Corporation 2021*c*). Microsoft Azure is on a pay-as-you-go basis, meaning customers pay for what they have been using. Additionally, for educational purposes or newbies who want to explore the platform, Azure offers a free 12-month account, including credit for 200USD that should be spent within 30 days (Microsoft Corporation 2021*a*).

The Text Analytics for Sentiment Analysis API provides two options:

- **Sentiment Analysis**. It returns a "positive", "negative", or "neutral" label and includes confidence scores at sentence and document-level.
- **Opinion Mining**. It provides more granular information that includes attributes of products or services. The `opinionMining=true` parameter must be included before calling the API.

It is required to set up a *Text Analysis* resource to start using the *Sentiment Analysis* feature where the *pricing tier* is selected (the option F0 is for the free pricing tier). The deployment process creates an API Key and an URL to connect to the API. With this information, the call to the API can be done through the command prompt window using cURL. However, Azure has available a client library for different programming languages, including Python. Listing 2.2 shows a code example in Python.

```
from azure.ai.textanalytics import TextAnalyticsClient
from azure.core.credentials import AzureKeyCredential

key = "<paste-your-text-analytics-key-here>"
endpoint = "<paste-your-text-analytics-endpoint-here>"

# Authenticate client
ta_credential = AzureKeyCredential(key)
client = TextAnalyticsClient(endpoint=endpoint, credential=ta_credential)

# Example
documents = ["I had the best day of my life. I wish you were there with
    me."]
response = client.analyze_sentiment(documents=documents)[0]
print("Document Sentiment: {}".format(response.sentiment))
print("Overall scores: positive={0:.2f}; neutral={1:.2f}; negative={2:.2f
    } \n".format(
    response.confidence_scores.positive,
    response.confidence_scores.neutral,
    response.confidence_scores.negative,
))
```

Listing 2.2: Azure code example in Python.

For the Sentiment Analysis feature, documents sent to API must be of a size not over 5,120 characters. Whether the document excites the limit of characters, this will not be processed. The number of documents allows to send is ten documents per request. In the case of the *F0* pricing tier (i.e., free plan), the rate limit is 100 requests per second and 300 requests per minute. If one of these conditions is violated, the API rejects the request and cause an HTTP 400 error (i.e., bad request) (Microsoft Corporation 2021*f*).

Related to data, privacy, and security for Text Analytics, the data send to the API may be temporally stored for up to 48 hours and then purged. However, the user can prevent this by including on the query the parameter `LoggingOptOut=False`. Additionally, in the Privacy Policy, it is stated that "*the data is controlled by the user [..]*" and not replicated to another

location (Microsoft Corporation 2021*f*).

Unfortunately, Microsoft Azure was not considered for the labeling process in this project since the amount of data planned to send to the API would exceed the number of requests available on the free account (5,000 requests per month). Moreover, the credit given would not be enough to complete the process (Microsoft Corporation 2021*b*). Additionally, as stated in the Transparency note for Sentiment Analysis, "*the machine learning model that is used to predict sentiment was trained on* ***product and service reviews****. This means the service will perform most accurately for similar scenarios and less for scenarios outside product and service reviews.*" As most of the tweets collected present a personnel opinion, this might show a low accuracy in the classification process (Microsoft Corporation 2021*d*).

2.9.3 Lexicon-based tools

This approach has been broadly used for opinion (or sentiment) classification. This algorithm attempts to classify the text according to the degree of polarity present within the text that most of the time it is applied to adjectives, adverbs, and some verbs and nouns that show a feeling/opinion (Zhang et al. 2011):

- **Positive polarity**: words encoded on a desirable state such as *good* or *great* adjectives.
- **Negative polarity**: words encoded on an undesirable state such as *bad* or *awful* adjectives.

Within a sentence, it is possible to compute the polarity score for each word which is called as *semantic orientation*, a measure of subjectivity and emotion within a text. In a nutshell, given a sentence S containing an entity e, opinion words in the sentence S are first identified by matching the opinion lexicon V. Then, the semantic orientation score is assigned to each word for the entity e. The semantic orientation score is assigned as follows (Taboada et al. 2011):

- For a positive word, the value of the semantic orientation score is +1.
- For a negative word, the value of the semantic orientation score is -1.

Afterward, all scores are summed up using the equation 2.4. In this equation, the reciprocal of $d(w_i, e)$ is used to assigned low weights to words far away from the entity e (Bhuta et al. 2014).

$$score(e) = \sum_{w_i : w_i \epsilon S \cap w_i \epsilon V} \frac{w_i \cdot SO}{d(w_i, e)} \tag{2.4}$$

where:

w_i is an opinion word.

V represent the lexicon, a set of opinion words.

S is the sentence that contains the entity (topic) e.

$dis(w_i, e)$ is the distance between the opinion w_i and the entity e.

$w_i \cdot SO$ is the score of the word w_i.

The advantage of using this approach is that must of the tools available are open source, no need for training data, and fast performance. Thus, the lexicon-based approach was used for the labeling process.

2.9.4 Survey on lexicon-based tools

There is a wide variety of free lexical resources for Python. Academic researchers had compared different lexicon-based tools where the most commons were VADER, TextBlob, and SentiWordNet. This section discusses each tool and compares the performance.

Valence Aware Dictionary for sEntiment Reasoning (VADER) is a lexicon and rule-based Sentiment Analysis tool able to detect whether a text has a positive, negative or neutral polarity, working perfectly on micro blog-like text such as tweets. In 2015 VADER was introduced in an academic report by Hutto and Gilbert. They found out that VADER performs as well as human labeling in social media texts. Thus, the benefits of VADER overcome traditional sentiment lexicons like The Linguistic Inquiry and Word Count (LIWC) tool.

SentiWordNet another tool for opinion mining, based on the *WordNext Lexicon*. The scores given are produced with a semi-supervised ML algorithm with values from zero to one, indicating the sentiment of a text (Al-Shabi 2020).

TextBlob is another library that includes text processing tasks for NPL. Moreover, it includes a sentiment property that returns polarity (decimal number from [-1,1] where -1 is negative and 1 is positive) and subjective (decimal number from [0,1] where 0 is objective and 1 is subjective). This is recommended for newbies on NPL as its simplicity and beginner-friendly (Williams et al. 2019).

Al-Shabi (2020) examined different lexicon tools on two datasets. In each test, VADER outperformed with 72% and 65% of accuracy scores. Table 2.15 shows a summary of these results. Bonta et al. (2019) had performed a similar work where VADER outperformed with an accuracy score of 77%, whereas Textblob and NLTK have 74% and 62%, respectively. Particularly, Sentiment Analysis on tweets has been carried out in other academic research works using VADER for labeling. Hence, the lexicon-based tool used for the labeling process in this project was VADER.

	Accuracy	
Dataset	(1)	(2)
VADER	**72%**	**65%**
SentiWordNet	53%	59%
sentiStrength	67%	58%
AFINN-111	65%	62%

Table 2.15: Al-Shabi's results on lexicon comparison.

2.10 DataCamp

As Natural Processing Language and Sentiment Analysis were entirely new topics, four courses were taken on DataCamp to complete this project. The access provided to these courses was given by the institute through the free classroom plan.

Founded in 2013 by Martijn Theuwissen, Jonathan Cornelissen, and Dieter De Mesmaeker, *DataCamp* is the first online platform focused on data science learning experience. All topics can be found in data science, statistics, and machine learning. No installation required—run as code runs from the browser. DataCamp provides different learning methodologies (DataCamp, Inc. 2021*a*):

- **Learn**. Interactive courses. There are also
- **Practice**. Quick daily challenges to enforcing what has been learned.
- **Apply**. Projects available to solve real-world problems.
- **Assess**. Test new skills and track process.

DataCamp offers different tracks according to learner's needs. There are two modes: skill tracks and career tracks. Skill tracks are a collection of domain-specific expertise courses (e.g., Importing & Cleaning Data course) whereas career tracks guide to the right path career (e.g., Data Analytics career path)(DataCamp, Inc. 2021*b*).

The course to learn more about Sentiment Analysis is called *Sentiment Analysis in Python* and was taken during May 2021 (during the intense project period). The duration of this course is roughly 4-hour length, including 16 videos and 60 interactive exercises for practicing (DataCamp, Inc. 2019). The course includes four chapters.

- **Sentiment Analysis Nuts and Bolts**. The basic structure of a Sentiment Analysis problem and exploration.
- **Numeric Features from Reviews**. Transform the text into a numeric form and consider a few complexities in the process.
- **More on Numeric Vectors: Transforming Tweets**. Additional complexities for social media data. Additionally, learn other ways to obtain numeric features from the text.
- **Let's Predict the Sentiment**. Employ logistic regression to predict the sentiment.

Other courses were taken from 21 June 2021 to 28 June 2021 to acquired more knowledge on NPL:

- Introduction to Natural Processing Language in Python.
- Advanced NPL with spaCy
- Feature Engineering for NPL in Python

Statements of accomplishment were granted after completing these courses, included in Appendix C.

Chapter 3

Exploration of the Data

3.1 Data Collection

The data used for the sentiment analysis on Covid-19 vaccines was extracted from the Twitter API, gathering all tweets and subsequent replies/conversations with the word "covid vaccine" or any related words/synonyms subjected to this concept. It was required to send an application to the Twitter Dev Team to get access to the API. As a researcher, on the 15th of March 2021, I applied to the *Academic research product*, explaining the project in detail. After eight days, Twitter sent the approval via email, providing a link to create a project and get keys and tokens to call the API. All support documents related are included in Appendix B.1. In a nutshell, Figure 3.1 shows the tasks performed in this stage.

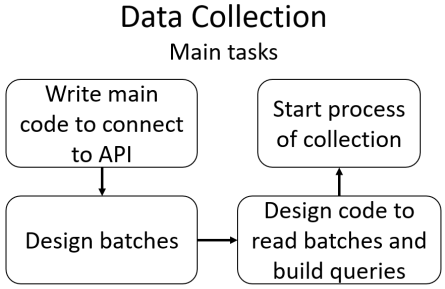


Figure 3.1: Tasks performed for Data Collection process.

The first task was writing a script file in *Python* to test the connection to the *Recent search* endpoint, collecting five tweets. Once it worked properly, the script was slightly changed to use the *Full-archive search* endpoint and collected 100 tweets as a second test process. Appendix A.2 includes the first version of the code created in March 2020 and modified in May 2021, available from *https://github.com/karla-cepeda/Dissertation/Code/Extras*. The new version of the code created in June and July 2021 is available in a Github repository from *https://github.com/karla-cepeda/Dissertation/Code*.

The next task performed was the design of the queries. The query created to filter the tweets had three important parts: the keywords, the place, and the usernames. Table 3.1 explains each part of a query. In a first trial performed in March 2020, tweets were collected, using the operator "place" to gather tweets from Ireland, returned a collection of fewer than 10,000 tweets. A second trial was performed in May 2021, increasing the number of tweets

when including Irish usernames related to media, government, and health in such a way to collect Irish tweets without the operator "place." Therefore, all users who have replied to these usernames are assumed to be residents of Ireland[1]. On this procedure, the queries collected roughly 30,000 tweets in total. However, to increase even more the number of tweets for the modeling stage, it was decided to collect any tweet related to "covid" and "vaccine" excluding Ireland with the operator "place" on negation format (i.e., -place: IE). This query collected more than 300,000 global tweets, exclusively used for the modeling stage with these conditions set. It is important to remark that in July 2021, the collection process was executed a third time due to modifications to the Python code. Unexpectedly, the number of Irish tweets increased by around 100,000 tweets. Despite this, global tweets were used for modeling as this dataset still contains more observations.

The rate limit for the Full-archive search endpoint is one request per second and 300 requests every 15 minutes as the python code called the endpoint to collect tweets created from January 2020 to 13 August 2021[2]. Roughly there would be 548x73[3]=3,836 requests. Consequently, there was a delay applied for every 300 requests to violate the rate limit. Additionally, another one-second delay was applied because the call was too fast, and in some cases, more than one request was called within a second.

The data was collected in 25 batches, depending on the query sent to the endpoint. Table 3.2 shows the name of the batches and a brief description, and what keywords are included. The batches were saved into a YAML file similar to a dictionary data structure in Python, where data is stored in a key-value format. The number of batches was largely due to the length of the query, as its limit size is 1,024 characters. Sending "vaccines," "pharmaceutical brands" (such as pfizer), and "vaccine names" (such as comirnaty from Pfizer) together exceed the length allowed. To this extent, these keywords were split into different batches to reduce the length of the queries. After collecting tweets, a sub-task extracted conversations by sending the `conversation_id` from each tweet collected from batches with the prefix "from-." All data collected was stored on JSON files and organized into folders to keep the data in a "raw" format as it contains special characters and emojis that were removed in the Data Preparation stage.

More parameters were added in the query to collect more information, and the response returned the following objects:

- **Tweet object**. All available data.
- **User objects**. This data was collected just from groups/entities and public figures. Other users were excluded and just collected a unique identifier feature that is located in the Tweet object.
- **Place object**. The country and city name were the only data collected (if applied as some tweets do not have a tagged location).

[1] In other words, it is assumed that Irish residents just reply to these usernames on the social media Twitter.
[2] The first collection for the interim report was done on 21 March 2021.
[3] Number of times the code was running again with a different query.

Query section	Description
Keywords	This section includes all possible synonyms regarding to "Covid-19 vaccine": • For **covid**: – covid, corona, coronavirus, covid19, covid-19, virus, sars-cov-2, sars cov 2, sarscov, pandemic – including hashtags: #covid19, #covid, #zerocovid, #covid19ireland, #covid_19, #pandemic, #covidireland • For **vaccine**: – vaccines, vaccinated, vaccination, dose, doses, injection – related to pharmaceuticals: pfizer, moderna, astra, astrazeneca, oxford, BioNTech, johnson, "johnson & johnson", "j&j" – related to (proposed) vaccine names: "mRNA-1273", bnt162b2, bnt162, AZD1222, NIAID, janssen, vaxzevria, comirnaty, sputnik, Gam-COVID-Vac, coronavac, sinovac, novavax, NVX-CoV2373, covaxin – including hashtags: #astrazeneca, #oxfordastrazeneca, #astrazenecavaccine, #pfizer, #pfizervaccine, #modernavaccine, #moderna, #johnsonandjohnson
Places	This section includes all possible tagged places related to the tweet. Operators included were: places cork, galway and dublin, country code IE and hashtags included #ireland, #dublin, #galway, and #cork.
Usernames	This operator was used to collect tweets from specific usernames (entities/groups and public figures): • **Media**. Tweets collected from users: rte (RTÉ), RTE_PrimeTime (RTÉ Prime Time), drivetimerte (Drivetime RTE), RTERadio1 (RTÉ Radio 1), Independent_ie (the Independent), NewstalkFM (NewstalkFM), IrishSunOnline (The Irish Sun), IrishTimes (The Irish Times), IrishTimesNews (Irish Times News), thejournal_ie (the Journal), irishexaminer (the Irish Examiner), and IsFearrAnStar (Irish Daily Star). • **Political Parties**. Tweets collected from users: sinnfeinireland (Sinn Fein), fiannafailparty (Fianna Fail), greenparty_ie (Green Party), nationalpartyie (National Party), finegael (Fine Gael), labour (Labour Party), and socdems (Social Democrats). • **Government and related**. Tweets collected from government department users or related: deptenterprise (Department of Enterprise, Trade, and Employment), citizensinfo (Citizens Information), welfare_ie (Department of Social Protection), csoireland (Central Statistics Office Ireland), merrionstreet (MerrionStreet - News from the Government). Public figure users included: MichealMartinTD (Micheál Martin), LeoVaradkar (Leo Varadkar), PresidentIRL (Michael D. Higgins - President of Ireland). • **Health and related**. Tweets collected from users: hselive (HSE), hpscireland (HSE Health Protection Surveillance Centre), and roinnslainte (Department of Health). Public figure users added to this category. Those are cmoireland (Dr. Tony Holohan - Chief Medical Office), paulreiddublin (Paul Reid - CEO of Health Service Executive), ronan_glynn (Dr. Ronan Glynn - Deputy Chief Medical Officer, Department of Health), and donnellystephen (Stephen Donnelly - Minister for Health).

Table 3.1: Main sections of the query sent to collect tweets to Twitter API.

Batch	Name	Description
1.1	covid_vaccine	This batch collects all tweets from Ireland that contain "covid" and "vaccine" keywords and the operator place.
1.2	covid_vaccine_c	This batch collects all tweets from Ireland that contain "covid" and "pharmaceutical and vaccine names" keywords and the operator place.
2.1	from_media	This batch collects all tweets posted by usernames related to the Irish media and contains "covid" and "vaccine" keywords.
2.2	from_media_c	This batch collects all tweets posted by usernames related to the Irish media and contains "covid" and "pharmaceutical and vaccine names" keywords.
2.3	to_media	This batch collected all replies to tweets from usernames related to the Irish media and contains "covid" and "vaccine" keywords.
2.4	to_media_c	This batch collected all replies to tweets by usernames related to the Irish media and contains "covid" and "pharmaceutical and vaccine names" keywords.
3.1	from_politicalp	This batch collects all tweets posted by usernames related to Irish political parties and contains "covid" and "vaccine" keywords.
3.2	from_politicalp_c	This batch collects all tweets posted by usernames related to the Irish political parties and contains "covid" and "pharmaceutical and vaccine names" keywords.
3.3	to_politicalp	This batch collected all replies to tweets from usernames related to the Irish political parties and contains "covid" and "vaccine" keywords.
3.4	to_politicalp_c	This batch collected all replies to tweets by usernames related to the Irish political parties and contains "covid" and "pharmaceutical and vaccine names" keywords.
4.1	from_gov_public_figures	This batch collects all tweets posted by usernames related to Irish politicians and contains "covid" and "vaccine" keywords.
4.2	from_gov_public_figures_c	This batch collects all tweets posted by usernames related to Irish politicians and contains "covid" and "pharmaceutical and vaccine names" keywords.
4.3	to_gov_public_figures	This batch collected all replies to tweets from usernames related to the Irish politicians and contains "covid" and "vaccine" keywords.
4.4	to_gov_public_figures_c	This batch collected all replies to tweets by usernames related to the Irish politicians and contains "covid" and "pharmaceutical and vaccine names" keywords.
5.1	from_gov	This batch collects all tweets posted by usernames related to Irish government departments and contains "covid" and "vaccine" keywords (the users have been enlisted in table 3.1).
5.2	from_gov_c	This batch collects all tweets posted by usernames related to Irish government departments and contains "covid" and "pharmaceutical and vaccine names" keywords.
5.3	to_gov	This batch collected all replies to tweets from usernames related to the Irish government departments and contains "covid" and "vaccine" keywords.
5.4	to_gov_c	This batch collected all replies to tweets by usernames related to the Irish government departments and contains "covid" and "pharmaceutical and vaccine names" keywords.
6.1	from_health	This batch collects all tweets posted by usernames related to health and contains "covid" and "vaccine" keywords.
6.2	from_health	This batch collects all tweets posted by usernames related to health and contains "covid" and "pharmaceutical and vaccine names" keywords.
6.3	to_health	This batch collected all replies to tweets from usernames related to the health and contains "covid" and "vaccine" keywords.
6.4	to_health_c	This batch collected all replies to tweets by usernames related to the health and contains "covid" and "pharmaceutical and vaccine names" keywords.
7.1	covid_vaccine_global	This batch collected all tweets related to covid vaccine and contains "covid" and "vaccine" keywords. Information collected from this batch will be used for the modeling stage to label Irish tweets.

Table 3.2: Batch names and description.

Figure 3.2 shows the collection process performed in Python. Table 3.3 summarizes the core technology used in this stage.

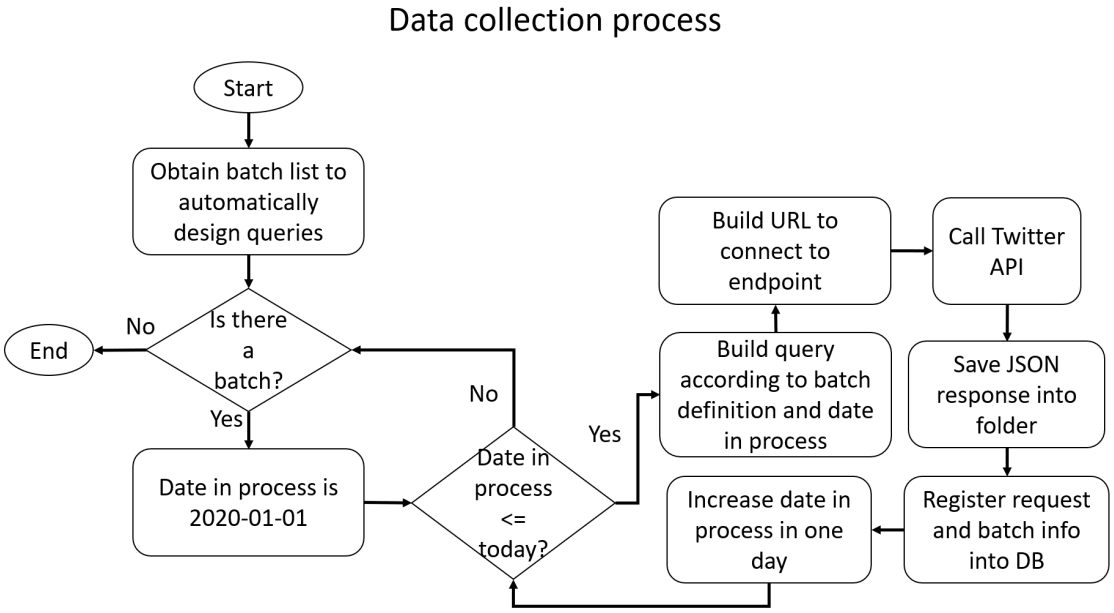


Figure 3.2: Data Collection process.

Core technology
Stage: **Data collection process**
Programming Language: **Python**
IDE: **Spyder and Visual Studio**
Libraries: • **request**. To call to Twitter API endpoint. • **json**. To save JSON response from Twitter API. • **yaml** To store API credentials.

Table 3.3: Core technology for data collection stage.

3.2 Data Preparation

In this phase of the project's life cycle, cleaning and normalization processes were executed on the collected data in the previous stage (i.e., Data Collection). The first step was to write a Python script to access the JSON files containing the raw data. After that, two functions were designed with Python to carry out cleaning and normalization on each tweet. Finally, this information was inserted into a local database called *Twitter* (further sections discuss the database structure). Figure 3.3 shows the main tasks performed during the data preparation stage.

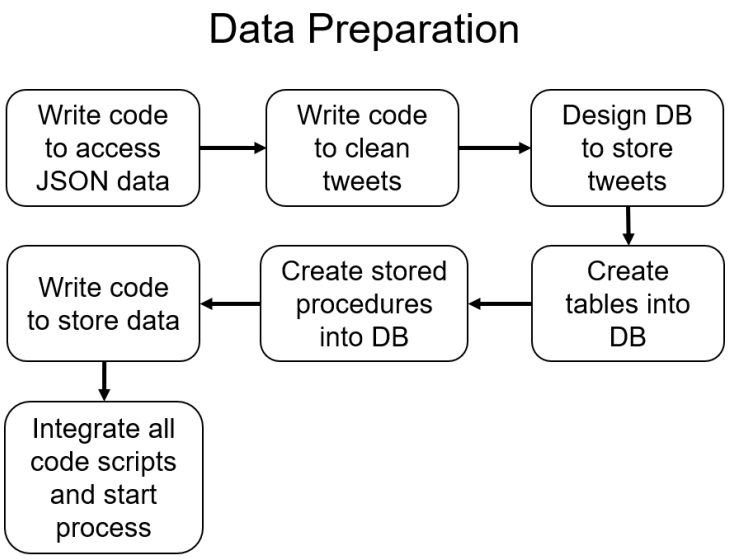


Figure 3.3: Tasks performed for Data Preparation stage.

Social media text is unstructured information, which means the cleaning process involves a series of tasks to remove irrelevant information (Wolff 2020). Some academic researches included emojis and emoticons for the modeling stage, although some showed slight improvement. In this project, emojis and emoticons were removed to try to deal with sarcasm. The cleaning process was based on the work of Talpada et al. (2019). However, additional steps were included from other sources (Joshi 2018, Shah 2020). The list below shows the steps applied to clean tweets:

- Expand contractions.
- Replace accents (I have found some tweets with accented words, decided to look for accents just in case).
- Remove mentions (text that starts with "@" mark).
- Remove URLs.
- Remove emojis and emoticons.
- Remove spaces on the left and right sides of each tweet.
- Remove double spaces (e.g., replace " " for " ").
- Remove special HTML characters such as
- Remove digits, including money, phone numbers, and percentages. Exceptions: covid-19 and sars-cov-2.
- Removing interjections.
- Remove other elements like emails, date, and time.
- Remove user Mentions (words that start with the "@" symbol are users mentioned in a tweet).

- Remove ASCII characters.
- Revise words with repeated characters.
- Revise spell of words.
- Translate slangs. Dictionary was taken from *https://www.noslang.com/dictionary/*.
- Replace censored word with asterisks such as f*ck => fuck.

One of the steps of normalization was the performance of Stemming or Lemmatization. Stemming reduces words into the same stem even if this does not have meaning, whereas lemmatization converts words into root and dictionary form. An example of this is "Studies" => "Studi" (stemming), "Study" (lemmatization). According to Balakrishnan and Lloyd-Yemoh (2014), on average, lemmatization methodology produces the best results on modeling. Therefore, for this section, lemmatization was applied in the normalization procedure using the library spacy. Other words removed were stop words like pronouns and articles (Angiani et al. 2016). According to Khafaji and Habeeb(2017), these words do not contribute to a Sentiment Analysis application as the lack positive or negative polarity. Angiani et al. (2016) suggested removing these words as "*they can lead to a less accurate classification.*" Consequently, stop words were removed using the library spacy. The steps followed to normalize the tweets are:

- Convert text into lowercase.
- Remove punctuation marks[4].
- Remove possession (i.e., "'s").
- Remove hashtag symbol and expand hashtags made up of two or more consecutive words.
- Remove English stop words such as "the", "is", and "of". Keywords used in the collection process were included in this list. Example of these are "covid", "vaccine", "pfizer", "moderna", and "astrazeneca".
- Lemmatization.
- One-word tweets with one-character length. This was removed as the lack of polarity and will be treated as empty tweets. One-word tweets will be explored in a further section.

Table 3.4 summarizes the core technology used in this stage.

[4]This step should be on the cleaning process for labeling process. The researcher has to fix it during the labeling process.

Core technology
Stage: **Data preparation process**
Programming Language: **Python**
IDE: **Spyder**
Libraries: • **ekphrasis**. To remove url, email, mentions, date, time, numbers with phones, money and percentage format. • **spacy**. To remove stop words and lemmatization process. • **preprocessor**. To remove emojis and emoticons. • **re** To remove specific patterns such as number excluding "covid-19". • **BeautifulSoup** To helper to remove HTML tags.

Table 3.4: Core technology for data preparation stage.

After a tweet was cleaned and normalized, an additional function was included to insert the tweets into a local DB, called *Twitter*. The Database Management System (DBMS) chosen is MySQL. Since the JSON files returned by the endpoints of the Twitter API were well-structured, this were used as a layout to design the structure of the DB. A normalization technique was implemented to verify this structure. However, no changes were applied as the tables satisfy the First normal form (1NF), the Second normal form (2NF) and the Third normal form (3NF). Figure 3.4 shows the structure of the database[5]. A brief description of the columns is enlisted below:

- **tweet**. This table contains all tweets in three versions: original text, cleaned tweets, and normalized tweets. The columns *cleaned_text* and *normalized_text* are used for labeling and modeling, respectably. The column *bacth_name* refers to the batch where a tweet came from. The column *key_name* refers to the part of the JSON response a tweet came from. A JSON response could return tweets in two sections: data and tweets. "data" contains the tweets looked up by the keywords sent in the query, whereas "tweets" contains referenced tweets such as retweets, quoted tweets, and replies to tweets. The columns *keywords* and *keywords_pharma* contain the keywords used to build the query sent to Twitter API. The columns *label* and *label_id* contain the sentiment related to the tweet. Tweets from the batch named *covid_vaccine_global*, the lexicon-based approach was used to get each tweet's sentiment, whereas tweets from other batches were classified using the ML algorithm built. The column *active* is used to digitally delete tweets to avoid tweets re-insertion during daily collection.

- **referenced_tweet**. This table contains all tweets and relation to other tweets. Additionally, it includes the type of tweet. For instance, if a tweet is a retweet type, the *tweet_id* that is in *referenced_id* is the root or parent, and the type is *retweet*.

[5]Diagram created on Graphviz Online.

- **user**. This table contains all users indicated on the batches such as media and political party users.
- **place**. This table contains all places only if `place_id` exists in tweet table.
- **log_file**. This table controls what file is processed. After downloading the JSON response, the file name and path are stored in this table. During the cleaning process, the column *preprocessed* controls what file has already been processed, mainly to control other JSON responses that are generated from the daily collection.

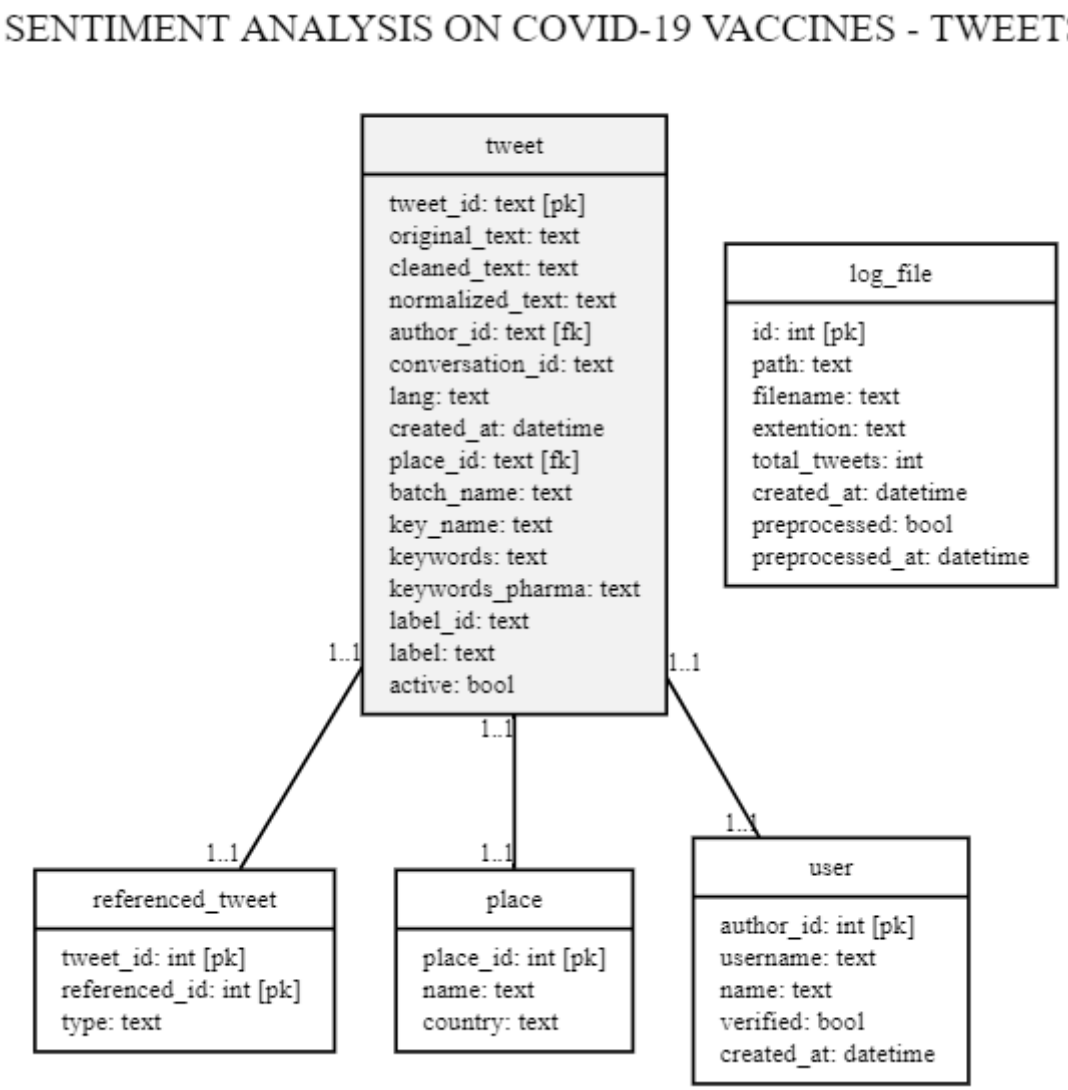


Figure 3.4: Entity Relationship diagram. Tables related to Twitter data collected.

In the poster presentation on the 15th of June 2021, the project review panel pointed out the lack of explanation on the tables *hashtag* and *tweet_hashtag*. After an analysis of these tables, it was decided to remove them from the database since these tables were not relevant for the research questions and goals of the project.

3.3 Description of the data

The dimension of the dataset is 155,950 observations from 1 January 2020 to 13 August 2021. This collection contains duplicated tweets as some of these tweets are retweets from different authors. The number of duplicated tweets identified is 16,972 tweets. Therefore there are 138,978 after removing them. The number of tweets increased unexpectedly due to some modifications to the code when collecting the whole conversation. Even though, the plan of using global tweets for modeling continued.

The batch *covid_vaccine_global* collected 319,627 tweets from 1 January 2020 to 13 August 2021. The number of duplicated tweets identified was 107,884 due to the presence of retweets from different authors. Therefore, there are 211,743 observations for the labeling and modeling stages.

In the following sub-sections, Irish tweets were used to create figures and plots. No further description of global tweets was performed as this is a secondary dataset used to train and test the model and subsequently label Irish tweets.

3.3.1 Type of tweets

On Twitter, there are different types of tweets which are:

- **Tweet**. This could be seen as the "root" or "parent", the starter of a conversation. In this report, to avoid confusion "tweet" term is used to refer to "all types of tweets" and the "post" term is used to talk about this type of tweet.
- **Retweet**. This is a re-posting of a tweet. This is just a quick way to share a tweet.
- **Quoted tweet**. It is a type of reply that displays the original tweet.
- **Replied to tweet**. A Reply message that is part of a conversation.

Figure 3.5 provides information on the proportion of Irish tweets collected and the type of tweets. Interestingly, the "Reply to" type has the largest number of tweets collected with 79.8%, whereas the lowest is "Retweets" with 1.1 percent. This make sense as for "Reply to," there were batches that collected the whole conversation and others that were designed to exclude"Retweets" when calling the API.

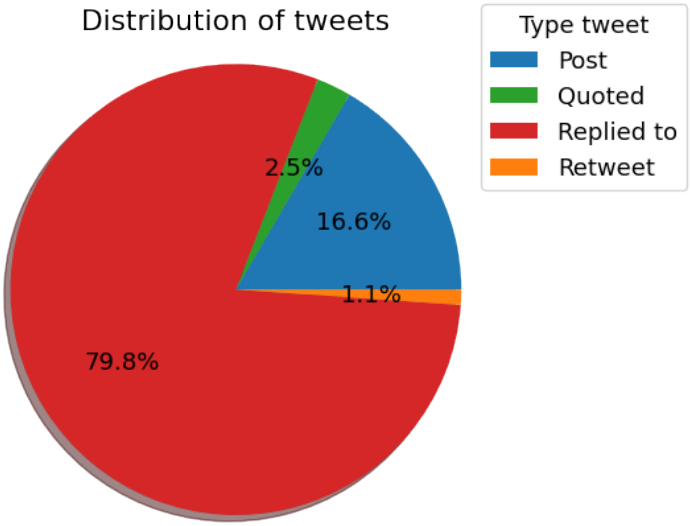


Figure 3.5: Type of tweets.

3.3.2 Batch categories

By merging some batches into generic batches, six categories were explored: *covid_vaccine*, *gov* (i.e., government), *health* (i.e., health department like HSE), *media*, and *politicalp* (i.e., political party). It is important to remember that some batches were designed to collect all tweets' replies by taking the `conversation_id` from every tweet. Therefore, when talking about conversations, this is referred to as a type of batch category. Additionally, these batches contain a sub-type with values "from," "to," or "None" related to the batches that contain usernames[6]. For more information, see section Data Collection.

Figure 3.6 shows the distribution of tweets. The media category has the largest proportion, with 39.6 percent. In second place is the health category with 22.5% and in third place is gov category with 24.3 percent. Interestingly, the *politicalp* category has the lowest number of tweets collected.

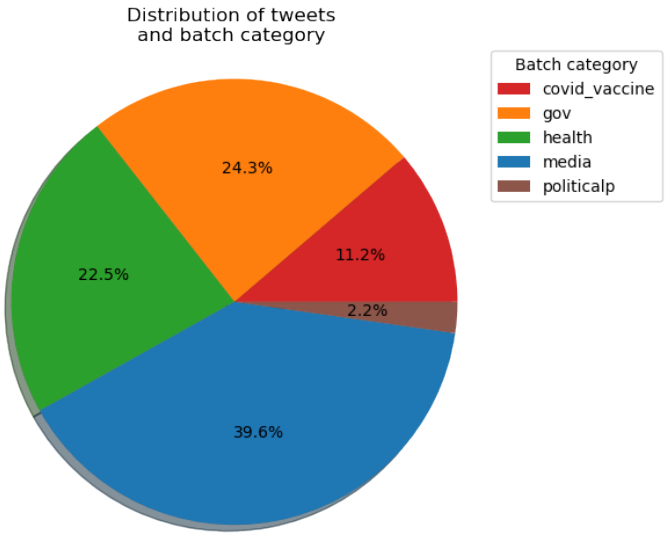


Figure 3.6: Distribution of batch categories.

[6]Since covid_vaccine batch does not contain a username list, the sub-type is "None".

Figures 3.7 and 3.8 show the proportion of tweets and batch categories on conversations and excluding conversations. The media category is still showing a large number of tweets. Figure 3.7, from the total of tweets, posted, the media category has the most significant proportion with 72.7 percent. In second place is the gov category with 18.1% of the total. The lowest proportion is the health category, with 56 tweets posted. By looking at Figure 3.8, the media category has 43.1% of the total, and the health category 27.1 percent. The lowest proportion is the politicalp category with 2.2% of the total tweets. A point to remark on these pie charts is the proportion of the health category as these users have posted 56 tweets since 1 January 2020. However, there had been 29,152 replies within the conversations.

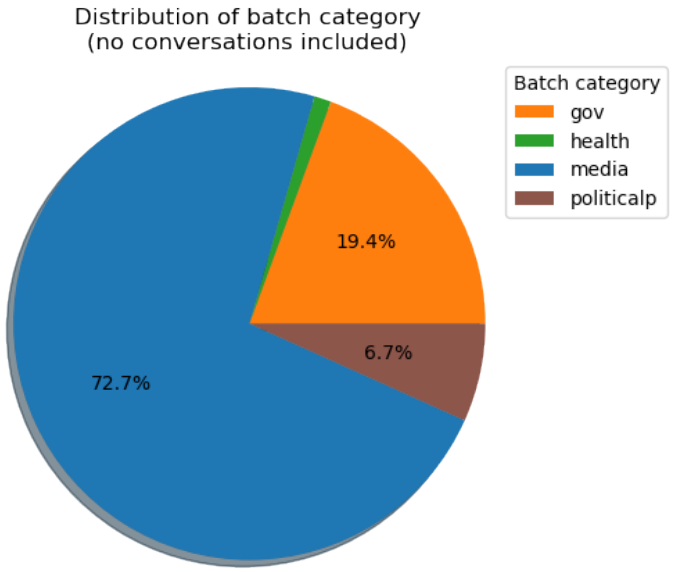


Figure 3.7: Distribution of batch categories and tweets.

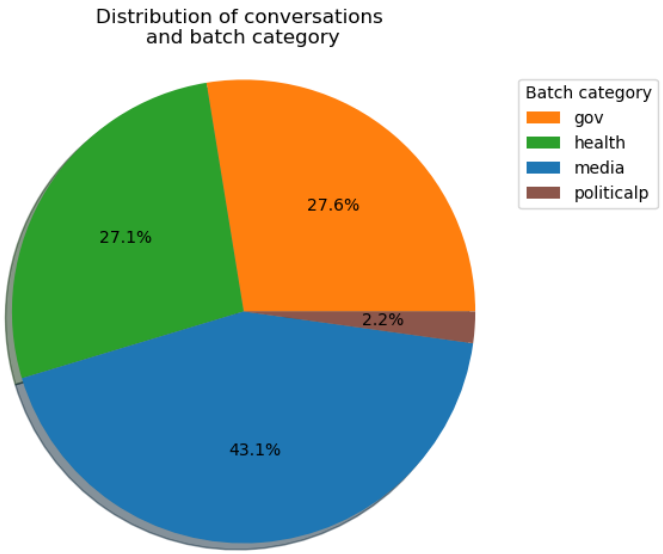


Figure 3.8: Distribution of batch categories and conversations.

3.3.3 Users within batch categories

As mentioned in previous sections, tweets from specific Irish usernames were collected. The primary purposes were to increase the number of tweets and analyze whether there is a feeling showed by entities or groups that communicate news related to covid and vaccines. Figure 3.9 shows the number of tweets posted and received per media usernames on Twitter. The plot shows that the *IrishTimes* user had posted 2,691 tweets since 1 January 2020, becoming the top among media users. In second place is the *Independent_ie* user with 1,786 tweets, and in the third position is *IrishExaminer* with 1,649 tweets posted so far. The proportion between "From user" and "To user" is unbalanced, showing that the Independent_ie user has the most significant ratio and number of replies.

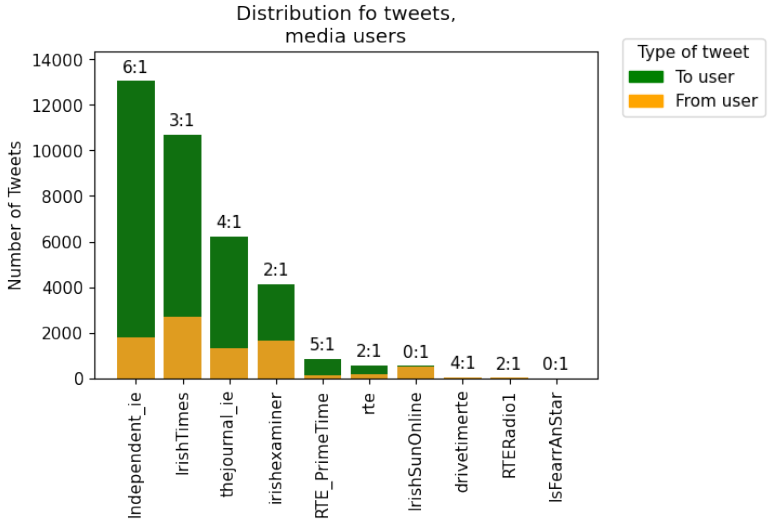


Figure 3.9: Number of tweets and media users.

Figure 3.10 shows the number of tweets posted and replies per political party user. The *FineGael* user has posted the most with 136 tweets in total, and the *NationalParty* user has the least with 12 tweets since 1 January 2020. It is noticeable the data on these users is unbalanced. In other words, the ratio *from:to* is not the same for each user. The username with a larger ratio and number of replies is the FineGael user with 919 tweets. The *SinnFein* and *FiannaFail* users are in second and third place, respectively.

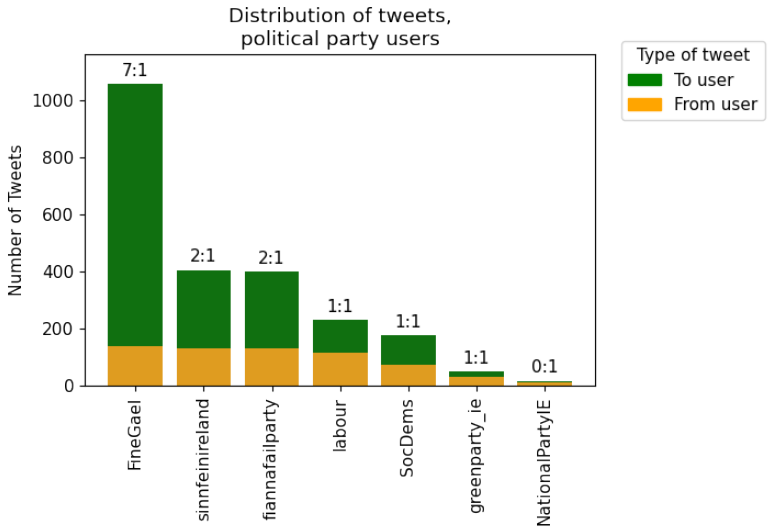


Figure 3.10: Number of tweets and political party.

Figure 3.11 shows two bar chart. The left side shows the number of tweets posted and received per government department users, and on the right one for politician usernames. Starting off the left side, the *MerrionStreet* user has the most significant number of posted tweets. This data is unbalanced as ratios have different values. The MerrionStreet user has the most significant ratio and with 805 tweets. By looking at the politician usernames, the *LeoVaradkar* user has posted the most, with 294 tweets since 1 January 2020. The data on these graphs is unbalanced, where the *MichealMartinTD* user has the most significant ratio. However, the LeoVaradkar user has the largest number of replies, with 4,727 replies. Noticeably, the public figure users have a more significant number of responses than government department users.

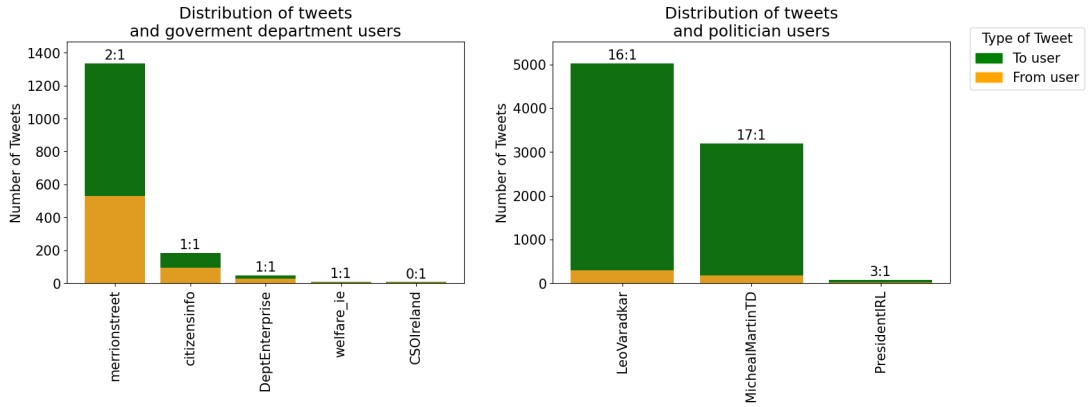


Figure 3.11: Number of tweets and government department users

Figure 3.12 on the left side shows the number of tweets posted and received per department users related to health, whereas on the right public figure users related to health. Starting off the left figure, the *HSE* user is first, which has posted 1,393 tweets. In second place is the *roinnslainte* (Department of Health) user with 621 tweets and final place the *HPSC* user with 148 tweets. The tweets distribution on these users is unbalanced, where the HSE user has the most significant ratio with 4,853 tweets. By looking at public figure users, the *DonellyStephen* user has posted the most significant number of tweets, with 474 tweets. In second place is *PaulReid* user with 249 tweets and third place the user *RonanGlynn* user with 116 tweets. Data from these users is unbalanced as "From:To" ratios have different values. Although the PaulReid user has the most significant ratio, the DonellyStephen user has received more tweets. It is interesting to notice that public figure users have a more significant number of responses than department users.

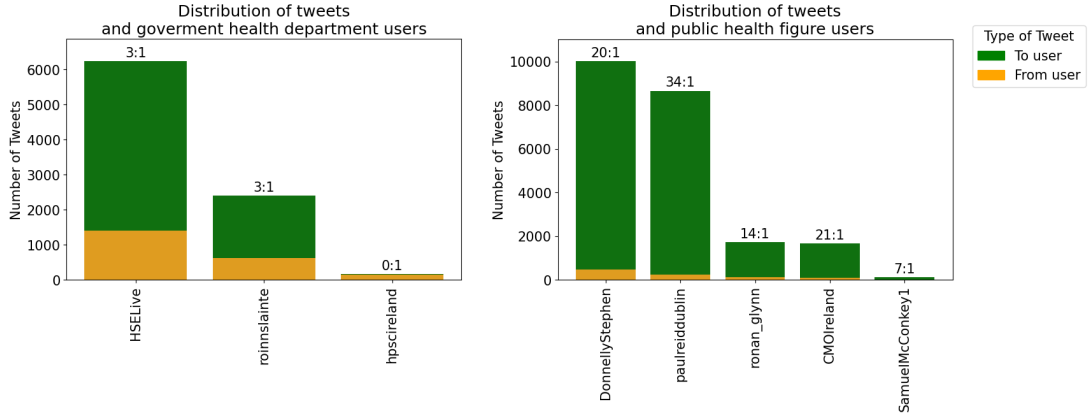


Figure 3.12: Number of tweets and health department users

Lastly, figure 3.13 shows the number of tweets posted among batches. The type of usernames that posted less were the government usernames, whereas the ones posted the most were the media usernames. The health department usernames started tweeting about Covid-19 vaccines around November 2020 and the political party usernames around May 2020.

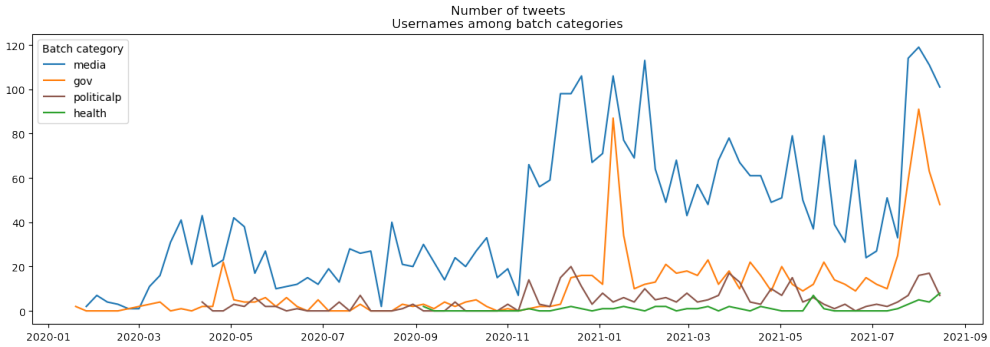


Figure 3.13: Number of tweets among batch usernames

3.3.4 Tweets

Figure 3.14 shows the length distribution (i.e., number of characters including blanks) and tweets before and after cleaning and normalization. The figure on the top shows the length distribution of the original tweets where the maximum number of characters is 2,345, the minimum is four, and the mode is 151. After the cleaning process, shown in the figure on the bottom left side, the distribution significantly changed where the basic statistics are: the maximum number of characters is 337; the minimum is one, and the mode is 74. On the figure on the bottom right side, after the normalization process, the distribution changed where the basic statistics are: the largest tweet contains 275 characters; the shortest tweet contains two characters; the most frequent length found is four characters. The distribution from all plots in Figure 3.14 is different since the cleaning and normalization process removed punctuation marks, mentions, hashtags, punctuation marks, etc. Normalization process was helpful for the modeling stage as it was a way to reduce features for the vectorization process. For more details on the cleaning process, see section Data Preparation.

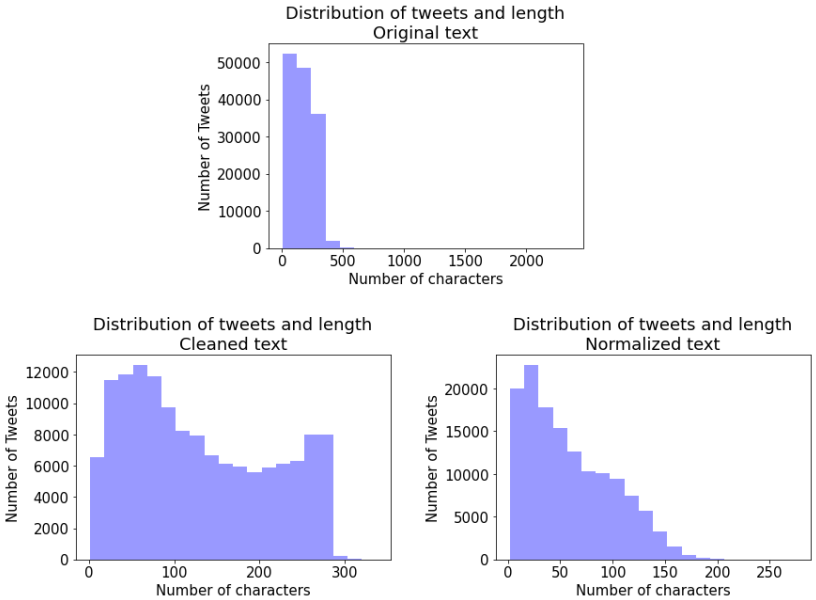


Figure 3.14: Distribution of the length and tweet.

Figure 3.15 shows box-plots from original, cleaned, and normalized tweets and the length (i.e., number of characters including blanks). Original tweets have various outliers where the extreme outlier has a value of 2,345 characters. Original tweets have a wider distribution and more significant variability in the bottom 50 percent. After the cleaning process, the outliers disappeared, and the range shrank. The box in the cleaned tweets overlaps the original tweets' box. Some outliers appeared on normalized tweets, but the spread is not extensive compared to the original tweets. Its box and variability bottom 50 percent shrank significantly.

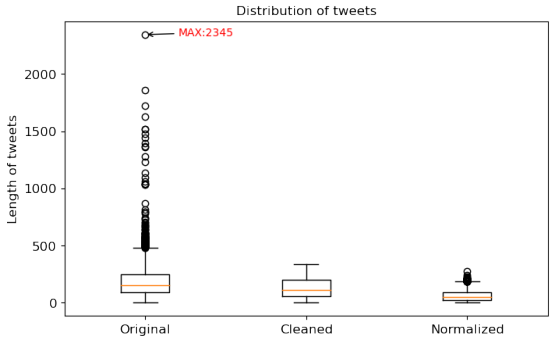


Figure 3.15: Distribution of length and tweet.

The distribution of words and tweets is shown in Figure 3.16, showing histograms from original, cleaned, and normalized tweets. The plot on the top shown the number of words over original tweets, where the basic statistics are: the maximum number of words within a tweet is 81 words; the minimum number is one word, and the mode is 14 words. After the cleaning process, the distribution changed showed in the plot on the bottom left side. The basic statistics for this are: the most significant number of words in a tweet is 78 words; the shortest remains 1, and the most frequent number of words is 11 words. Finally, the bottom right plot shows how the distribution of words has changed after the normalization process: the most significant number of words in a tweet is 48; the shortest remains one, and the mode is three words. The distributions on the top and bottom right plots are primarily different as normalized tweets do not contain stop-words, keywords and punctuation marks.

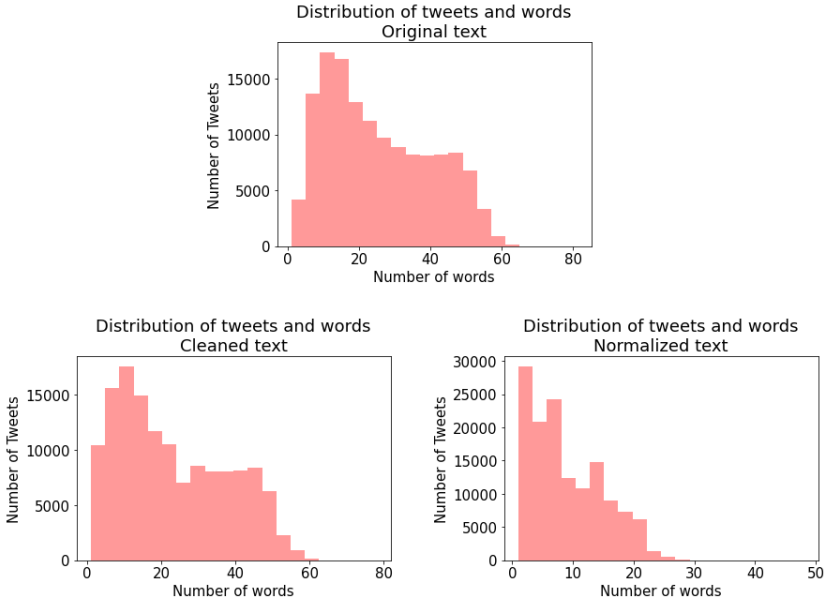


Figure 3.16: Distribution of words and tweet.

Figure 3.15 shows box-plots from original, cleaned, and normalized tweets and the number of words. Original tweets have two outliers, where the extreme outlier has 81 words. After the cleaning process, there are three outliers, and the range and media have slightly decreased. On normalized tweets, more outliers appeared, but the spread and variability bottom 50% has shrunk significantly. Normalized tweets have the lowest box and barely overlap the original and cleaned tweets. The range and boxes in original and cleaned tweets have slightly the same shape and size, a wider distribution, and a greater variability bottom 50% than normalized tweets. The box in cleaned tweets overlaps the original tweets' box.

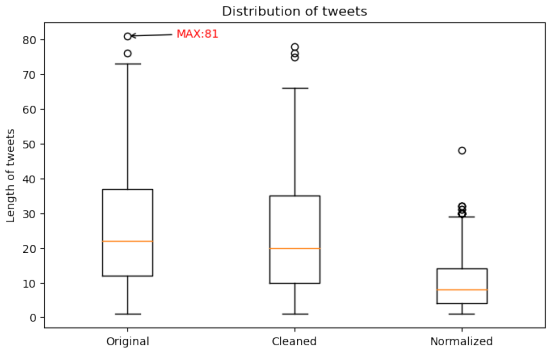


Figure 3.17: Distribution of words and tweet.

Figure 3.18 shows the top 20 words on the bag of words from cleaned and normalized tweets. Bag-of-words (BoW) is a simple and flexible approach to extract features from the text, describing the frequency of unique words in a text or collection of documents (Brownlee 2017). The BoW is used on the following plots to describe the dataset and the frequency of the words in cleaned and normalized tweets. Back to Figure 3.18, the plot on the left side shows the frequency of words within all cleaned tweets where the most frequent word is the article *the*. Among these, three words stand out: *vaccine*, *covid*, and *people*. This is important to notice as, in previous sections, many tweets came from conversations. Most of these tweets might not include the keywords defined in the collection process as these were downloaded with the parameter `conversation_id` which belongs to the tweet containing the keywords covid and vaccine[7]. Therefore, it is a good sign that *vaccine* and *covid* are still in the top 20 on cleaned tweets. On the left side, the bar chart shows the frequency of words within the normalized text. As pointed out by the project review panel in the poster presentation on the 15th of June 2021, and by the nature of the normalized tweets, keywords were not included in this plot. In this case, the most frequent word is *people*, whereas *need* and *year* are in second and third place, respectively. Figure 3.19 shows a word cloud image, where the size of the word represents the frequency. This word cloud was made up of normalized tweets text, standing out the word *people*.

[7]Or any synonym defined in the collection process.

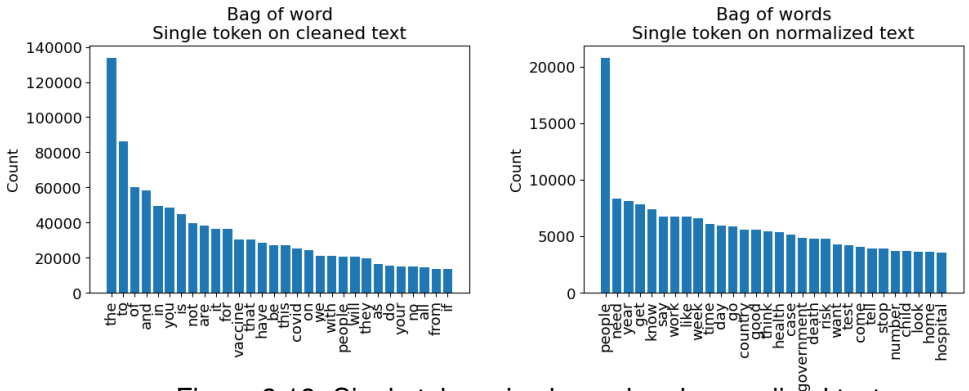


Figure 3.18: Single tokens in cleaned and normalized text.

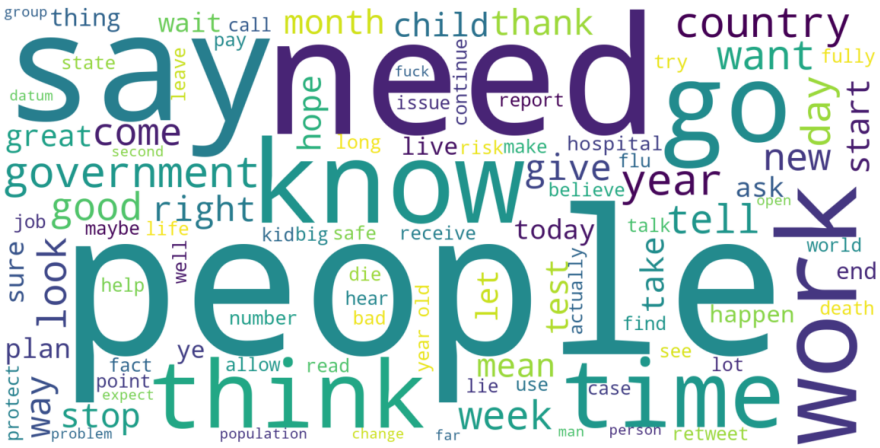


Figure 3.19: Word Cloud from normalized tweets.

Figure 3.20 shows the top 20 words on the paired bag of words[8] from cleaned and normalized tweets. On the left side, the plot shows the frequency of words within all cleaned tweets where the most frequent pair is *of the*, whereas in third place is *covid 19* and in fourth *the vaccine*. On the other hand, the plot on the right side shows that the top 3 pairs are: *year old*, *people not*, and *long term*.

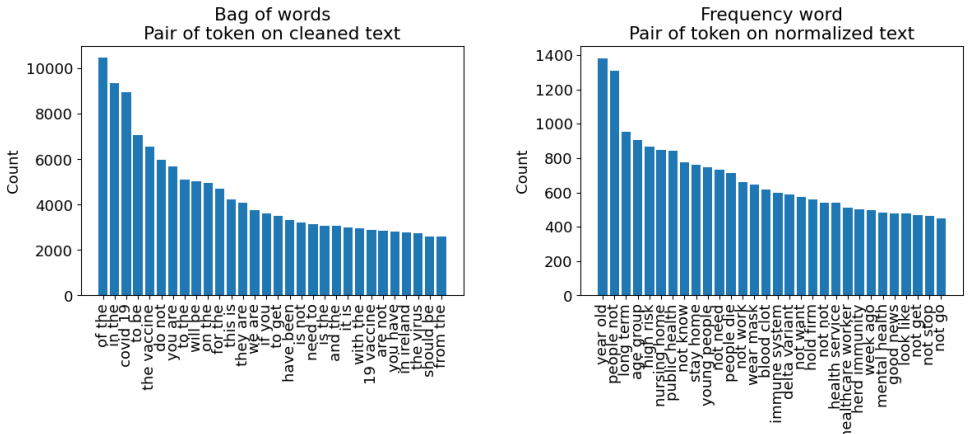


Figure 3.20: Paris of tokens in cleaned and normalized tweets.

[8]When a sentence is separated into two words.

Figure 3.21 shows the number of tweets over time, from 1 January 2020 to 13 August 2021. The line plot shows no trend and no constant variance. There were few tweets during 2020, but the number of tweets and variety started increasing after November 2020. The top 5 dates are:

- **2021-08-12** with 1,968 tweets (marked on graph). Possible trigger event: "*The overall vaccination program is in 'the final leg,' according to HSE chief executive Paul Reid, with 90 percent of adults have received at least one dose and 80 percent fully vaccinated.*" (Cullen & Hilliard 2021).
- **2020-12-26** with 1,527 tweets. Possible trigger event: "The first batch of the Pfizer BioNTech Covid-19 vaccine has arrived in Ireland with initial vaccinations expected to be administered on Wednesday." (RTE 2020*a*).
- **2021-04-06** with 1,462 tweets. Possible trigger event: "Minister for Education Norma Foley defended the Government's decision to change the vaccine roll-out schedule to an aged-based system stating it was "not a value judgment on any given profession", as teachers' unions continued to call for their members to be prioritized." (RTE 2020*c*).
- **2021-04-14** with 1,431 tweets. Possible trigger event: "Ireland was set to receive 545,000 additional doses of the Pfizer–BioNTech COVID-19 vaccine from April to June as part of a wider EU agreement." (O'Regan & Collins 2021).
- **2020-12-29** with 1,345 tweets. Possible trigger event: "A 79-year-old woman became the first person in the Republic of Ireland to receive the Pfizer/BioNTech COVID-19 vaccine at St. James's Hospital, Dublin." (Dwyer 2020).

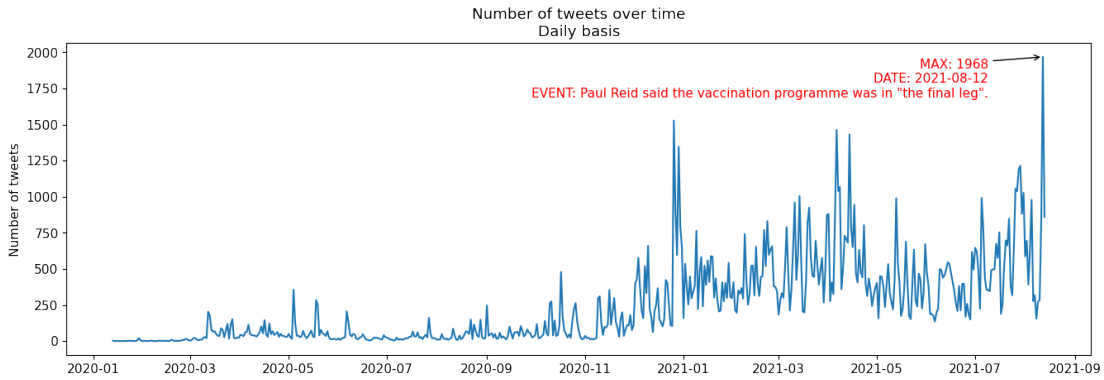


Figure 3.21: Daily tweets over time.

Figure 3.22 shows the number of tweets over time but segmented by type. The noise in Reply tweets type (third plot) is somewhat similar to the total number of tweets (bottom plot). This could be explained by the highest proportion of this type seen in Figure 3.6. The rest of the plots have different variances.

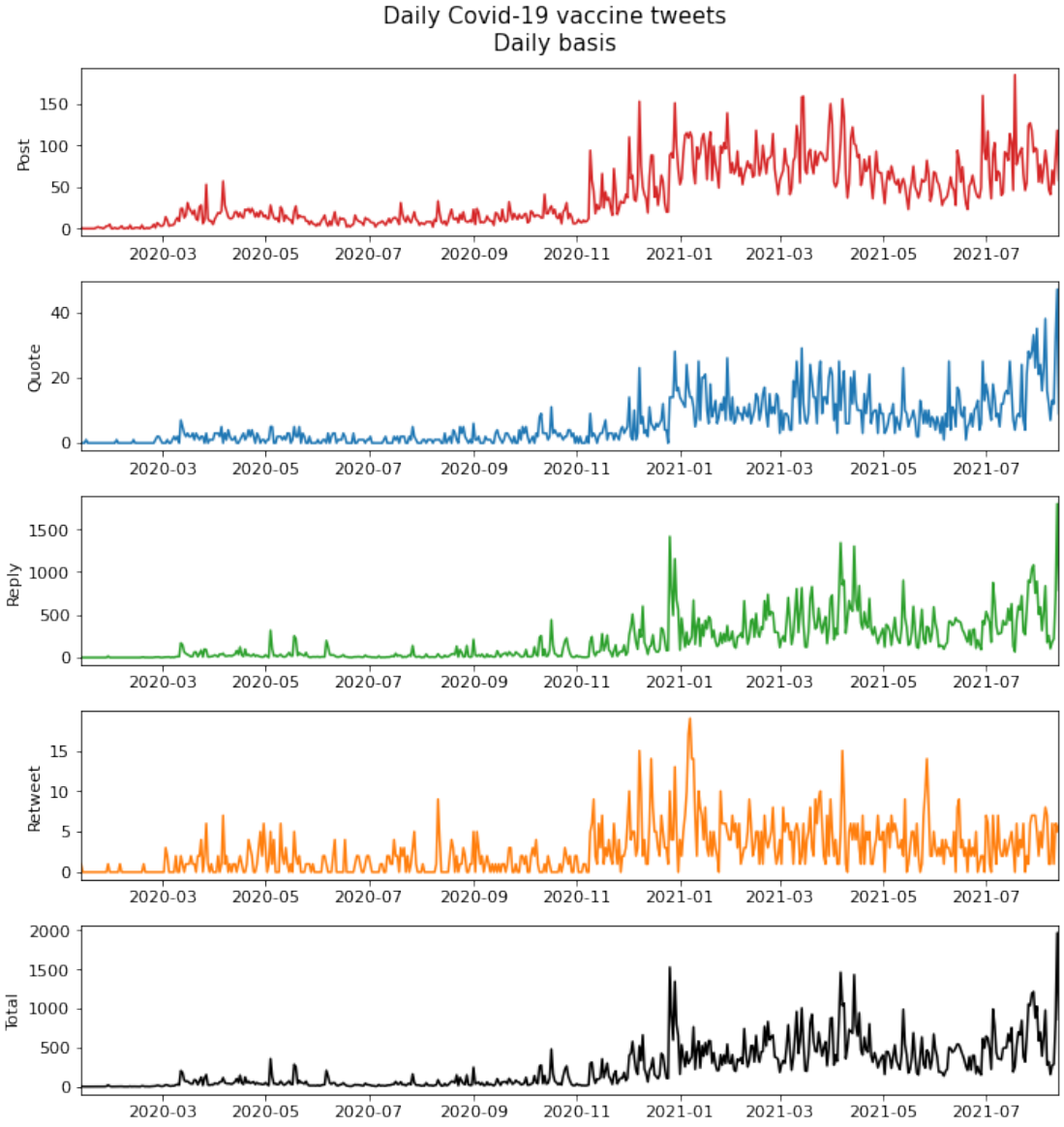


Figure 3.22: Daily tweets over time and type.

Figure 3.23 shows tweets weekly. The first week of August 2021 has the most significant number with 6,992 tweets, and the lowest is in the first week of January 2020 with just three tweets. Figure 3.24 shows the number of tweets monthly. July 2021 has the most significant number of tweets with 19,636, whereas January 2020 is the lowest with 37 tweets. Quarterly, the Q2 2021 has the most significant cumulative number with 41,151 tweets, whereas the lowest corresponds to the Q1 2020 with 1,747 tweets, shown in Figure 3.25.

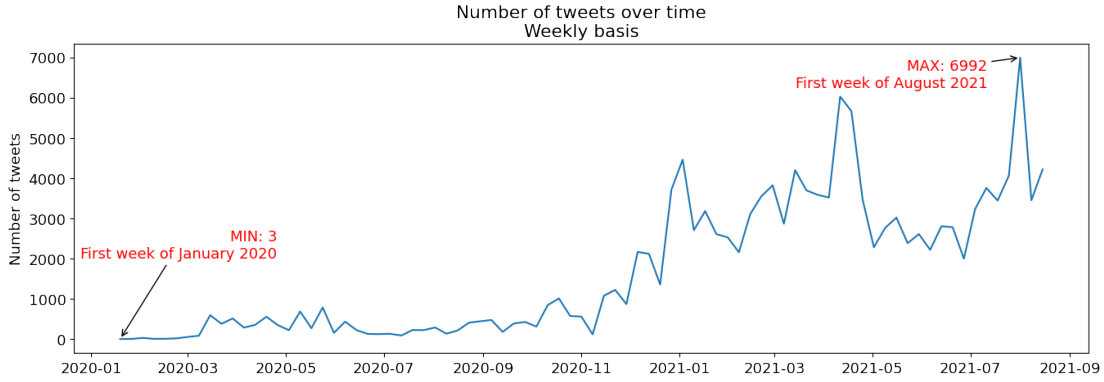


Figure 3.23: Weekly tweets over time.

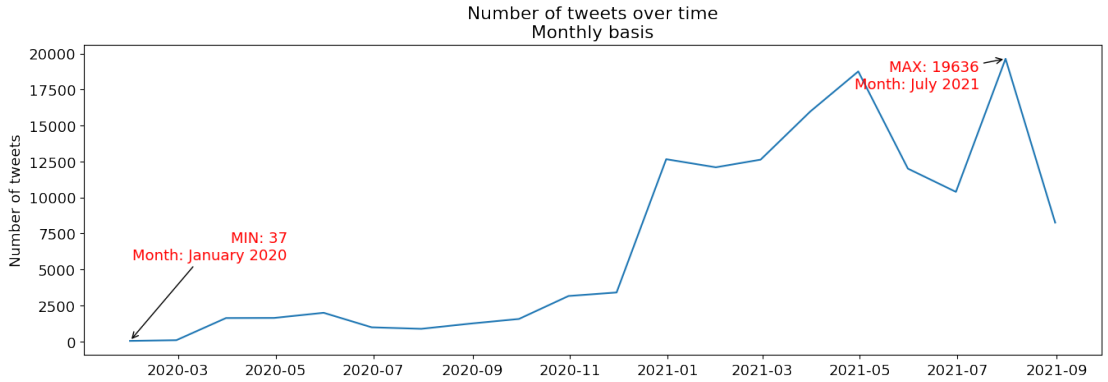


Figure 3.24: Monthly tweets over time.

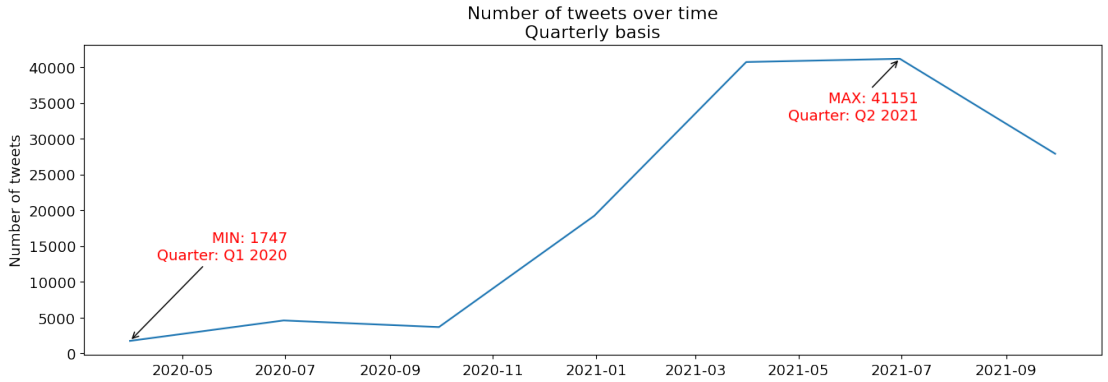


Figure 3.25: Quarterly tweets over time.

3.3.5 One-word tweets

Tweets with one word are analyzed in this section to decide whether include these words in the modeling process. Figure 3.26 shows the length and the number of one-word tweets. The basic statistics are: the longest word has 16 characters, the minimum length is two characters, and the mode is 4. There are 230 tweets with two-character length. Most of the words shown in Figure 3.27 do not make sense (i.e., lack of meaning). There are 995 one-word tweets with a third length long. However, some of these words have a sentiment like *bad*, *lie*, and *joy*. Therefore, one-word tweets of length two or less were digitally removed from the DB (i.e., `Active = False`). Although this analysis was performed on the Irish tweets, this is also applied to global tweets as both datasets were under the same cleaning and normalizing processes and belong to the same domain, *covid vaccines*. Figure 3.28 shows a word cloud made up of one-word tweets text (no words with two-length included), standing out the words *thank*, *fuck*, *good*, and *yes*.

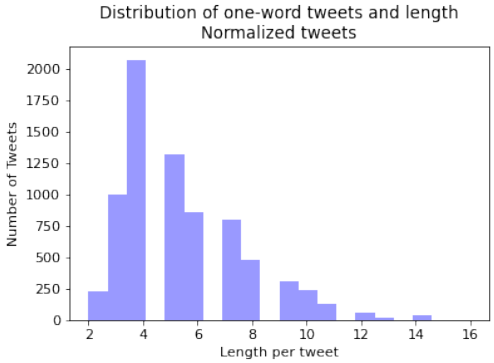


Figure 3.26: Number of tweets over time and type.

```
There are 230 tweets with one word one character
['no' 'dr' 'uh' 'ya' 'ah' 'ck' 'go' 'mo' 'el' 'de' '19' 'wo' 'ye' 'ar'
 'fl' 'hi' 'si' 'en' 'oh' 'jr' 'er' 'ti' 'cc' 'xx' 'pm' 'aw' 'co' 'qi'
 'al' 'ab' 'sh' 'gm' 'ff' 'eq' 'ad' 'ai' 'um' 'et' 'pi' 'ex' 'fo' 'fr'
 'la' 'ja']
```

Figure 3.27: Two-length word list.

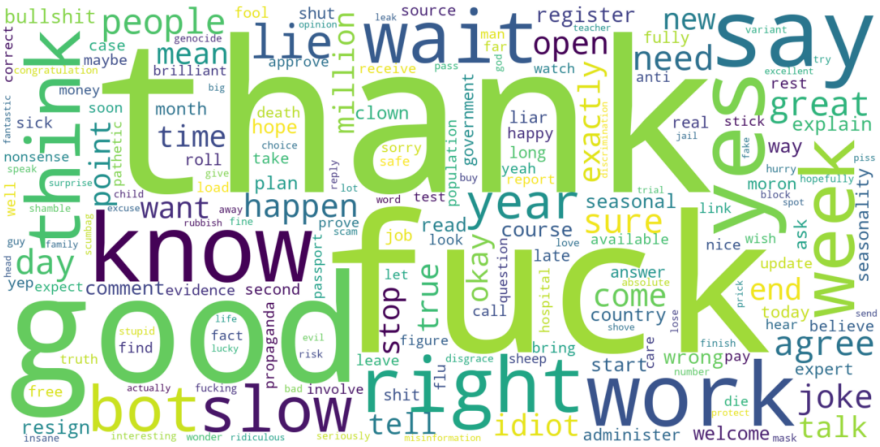


Figure 3.28: Word Cloud from one-word tweets.

3.4 Ethical considerations

The[9] tool used to collect all public tweets from the social media Twitter was the developer product *Twitter API*. After sending the application and then approved by the Twitter Dev Team, automatically the applicant is accepting the *Developer Agreement*, an agreement made between the developer/researcher and Twitter to conduct the access and use of the data, and "to keep Twitter's public conversations safe and healthy". Across all of the products, Twitter maintains strict policies and processes to assess how developers are using Twitter data, and restrict improper use of this data. When these policies are violated, actions are appropriately taken, which can include suspension and termination of access to Twitter's API and data products. Therefore, the main ethical considerations for this project are based on the *Developer Agreement* from Twitter, which are:

- Privacy
- Security
- Anonymization and/ or Potential for Identification of Individuals
- Property and Ownership of Data

As the research is targeted into Europe, more precisely in the Republic of Ireland, all the information retrieved is under the protection of the Twitter International Company (TIC) which offices are located in Ireland. As I am extracting Twitter data from Ireland, additionally I have to embrace the Twitter Controller-to-Controller Data Protection Addendum in addition to the General Data Protection Regulation (GDPR) privacy and security law.

Right after receiving the approval from Twitter to gain access to the Twitter API, the *Ethical Approval Application Form* was sent to the *Ethics Committee* from Dundalk Institute of Technology (DkIT) on the 23rd of March 2021 in which it was stated all ethical implications related to the project. On the 21st of April 2021, the project was considered by the *Ethics Committee* and ethical approval was granted.

The last collection extracted 155,950 tweets from the 1st of January 2020 to the 13th of August 2021. A daily collection is performed that is triggered every day at 23:50 in a local machine. This daily collector will be switched off by the end of September 2021. Related databases will be deleted as well.

In the following subsection, the most important ethical considerations undertaken on this project are described, based on the *Developer Agreement* (Twitter, Inc. 2020*a*).

[9]Most of the text in this section was taken from the 2nd Continuous Assessment, Ethics in Data Analytics module.

3.4.1 Privacy

When using the service on Twitter, the user consents to the use and collection of their information as stated in the Twitter Terms of Service (Twitter, Inc. 2020*d*). Regarding tweets, when a user posts (i.e., when *tweet*) public content, by default this information is disclosed by Twitter to facilitate the fast global dissemination of Tweets around the world. In other words, all the data that it is possible to collect through the Twitter API is classified as public and has implicit consent by the user to be utilized. Protected Tweets and Direct Messages are labeled as non-public, therefore there is no access to such information. Twitter restricts the use of these tweets in some use cases enlisted below:

- No display or share content derived from Twitter (unless explicitly approved).
- No conduct surveillance on users or content.
- No gathering knowledge to investigate or tracking users or content.
- No conducting or providing analysis or research for any unlawful or discriminatory purposes.
- No monitoring sensitive events, including protests, rallies, or community meetings.

3.4.2 Security

As Twitter has stipulated on the *Developer Agreement*, the data extracted from the Twitter API must not be shared with a third party, including any token, key, password, or other login credentials related to the product. Additionally, it is required to use industry-standard security measures to prevent unauthorized access. This includes:

- Any login credentials to Twitter API should not be sorted in shared repositories. It is recommended to have a central file for this such as a YAML file and stored in an external device.
- Regenerate Twitter API keys and tokens frequently. If keys are compromised, it is suggested to immediately regenerate them on the developer portal.

Regarding the safeguarding mechanisms, the following was set up to achieve security:

- The data is stored in a Database Management System (DBMS) with a strong password. The DBMS chosen is MySQL.
- No access to the database (DB) is provided, in the only except the supervisor or the project review panel required access to the data in the only case for project assessment. The access to the data would be *readable* (i.e., no allow to modification).

3.4.3 Anonymization and/ or Potential for Identification of Individuals

What is labeled as public information is the data available in the Twitter API. Regarding user's information, examples of available information are shown in table 2.8, such as username, and date when the profile was created. Confidential information such as age, full

name, phone number, and other personal information is not accessible. However, users have the option to use their real name on the field name (defined on profile, not necessarily a person's name) or username (i.e., screen name or alias). Thus, the information that would be collected related to users is their unique identifier (i.e., user ID) and just collect all user's information available from the username that represents a group or entity, that could include:

- A government department
- A political figure
- A mass media company

This action will prevent the potential re-identification of users, as there could be cases in which a tweet could be associated with a username. For instance, a username "johnsmith" has posted a tweet, then somebody read this tweet and states that their neighborhood, whose name is "John Smith", has tweeted an "unappropriated" opinion that leading to an inaccurate matched as there are thousands of "John Smith" around the world.

3.4.4 Property and Ownership of Data

Users are owners of their tweets. However, implicitly a consent by the user is given to Twitter to distribute public information to anyone who has access to the Twitter API (in this case) (Twitter, Inc. 2020*d*). In this context, it is not allowed to modify content, except for cleaning and modeling purposes.

Chapter 4

Design and Implementation

4.1 Labeling process

The labeling process was discussed in Section 2.8, where VADER was the lexicon tool selected due to its higher accuracy score comparing to other libraries. The library *nltk* was used in this project, the module *nltk.sentiment.vader*, that contains the class *SentimentIntensityAnalyzer*. This returns four values:

- **pos**: The probability of the sentiment to be positive. It takes a value between [0,1].
- **neg**: The probability of the sentiment to be negative. It takes a value between [0,1].
- **neu**: The probability of the sentiment to be neutral. It takes a value between [0,1].
- **compound**: A normalized score that computes the sum of "positive", "negative", and "neutral" ratings. It takes a value between [-1, 1] where -1 is very negative, 1 very positive and 0 is neutral.

In several academic research works that analyzed the sentiment in tweets, the compound score is the value used, and typically its threshold is 0.05. Thus, for this project, the following was used for the labeling process:

- **Positive**: compound score greater or equal to 0.05
- **Negative**: compound score lower or equal to 0.05
- **Neutral**: compound score between -0.05 and 0.05

After a tweet is classified, the *label* and *label_id* values were saved into the DB where "positive" is id "1", "negative" is "2", and neutral is "0". Table 4.1 shows a summary of the core technology used in this stage.

Core technology
Stage: **Labeling process**
Programming Language: **Python**
IDE: **Visual Studio**
Libraries: **ntlk**.

Table 4.1: Core technology for labeling stage.

Figure 4.1 shows in a simple way the labeling process performed on Python.

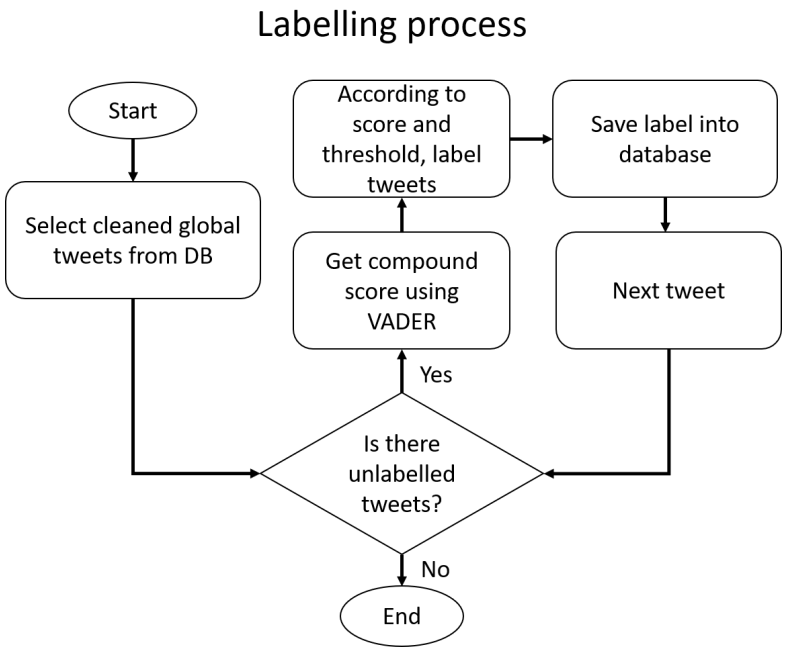


Figure 4.1: Data Collection process.

A Python script was coded to label all global tweets. The number of tweets labeled using the lexicon approach was 319,627 tweets related to *covid* and *vaccine* (from the batch covid_vaccine_global). Figure 4.2 shows the proportion of sentiment where *positive* is predominant.

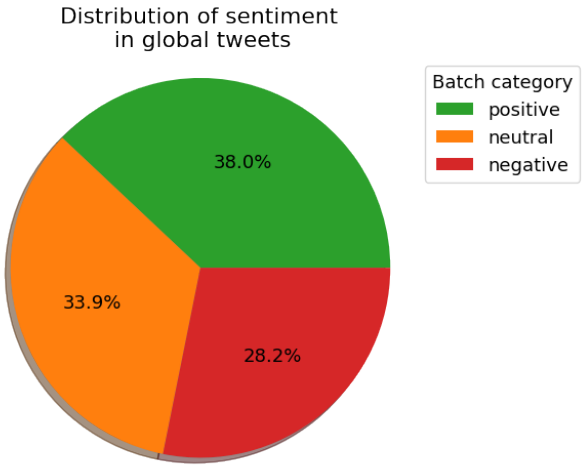


Figure 4.2: Global sentiment on Covid-19 vaccine.

As mentioned in Section 3.2 (Data Preparation), there was confusion between cleaning and normalization processes when dealing with the punctuation marks, where the removal was executed in the normalization processes instead of in the cleaning process. Before getting the compound score using VADER, a simple procedure was carried out to remove punctuation marks from the cleaned tweets.

4.2 Based model

As discussed in Section 2.7 (Survey on sentiment classification techniques), SVM was implemented to design the sentiment classifier. The data was downloaded from the local DB and used normalized tweets. To start training the model, first off, the unstructured data from tweets was vectorized. For feature construction, n-grams are the most common approach in Sentiment Analysis, a combination of words in a given window (Raghunathan 2020). As an example, taking the text "I love this," if n=1, this will take single tokens from the sentence example like "I," "love," and "this," whereas n=2 will take pairs of tokens such as "I love" and "love this." For feature weighting, there are two typical approaches (Paul 2020):

- **Term Frequency**. This is the number of times a word appears in a document divided by the total number in all documents, formula is shown in equation 4.1, where d is a document (i.e., tweet), and t is a word within the document.

$$w(d,t) = TF(d,t) \tag{4.1}$$

- **Term Frequency-Inverse Document Frequency**. This combines TF and IDF to weight features. IDF formula is given in 4.2, the log of the documents divided by the number of documents that contain that world. This combination results in the equation, measuring how much important a word is within a document. 4.3.

$$IDF(t) = log\left(\frac{N}{n}\right) \tag{4.2}$$

$$w(d,t) = TF(d,t) \times log\left(\frac{N}{n}\right) \tag{4.3}$$

Elouardighi et al. (2017) compared several combinations of n-grams (mainly uni-grams, bigrams, and combination of both) and weights (TF and TFIDF) on various ML algorithms including SVM. Their results showed that test 2 (uni-grams with TFIDF) gave a better accuracy score for the SVM classifier than the other test. Therefore, for the model, uni-grams and TFIDF were taken to vectorize the tweets. The submodule *sklearn.feature_extraction.text* and class *TfidfVectorizer* were used to construct the features and weights for vectorization.

The SVM algorithm has different kernels to transform the data, which has been discussed in Section 2.7 (Survey on sentiment classification techniques). A linear kernel is recommended as, according to Hearst et al. (1998), "*this provides a good generalization accuracy and is fast to learn*". Joachims (1998) studied linear and non-linear SVM algorithms, finding out an insignificant benefit comparing to a linear SVM. The *sklearn* library includes different SVM estimators with a linear kernel such as *svm.SVC*, *svm.LinearSVC*, and *linear_model.SGDClassifier* (when parameter `loss='hinge'`). In the documentation of the class *svm.SVC*, it is stated that with more than 10,000 observations its performance will be affected due "*the fit time complexity is more than quadratic with the number of samples*" (sklearn 2021*i*). The class *linear_model.SGDClassifier* utilizes a stochastic gradient descent optimizer to converge on a solution, and this is scalable for even millions of data rows, making it faster and may generalize better comparing to *svm.LinearSVC* (sklearn 2021*e*).

An example code available in the sklearn website is used to implement the based model using *linear_model.SGDClassifier* (sklearn 2021*j*, Bronchal 2017).

Additionally, for this project, a pipeline was used to chain the vectorizer and the SVM classifier. This has the following purposes: encapsulation, as functions `fit` and `predict` can be called once; tuning hyper-parameters easily; and, enforce an order of steps in model (sklearn 2021*d*). The class used for this was *sklearn.pipeline*.

Figure 4.3 shows the design of the prototype. In general, it includes the following five steps:

- Step 1. Removing short tweets. As seen in Section 3.3 (Description of the data), most of the one-word tweets with one and two-length were not relevant and may have a lack of sentiment, these tweets were removed. In total, there were 319,627 global tweets. After removing one and two-length tweets, the number of tweets for training was 319,043 tweets.
- Step 2. Remove duplicated tweets. 107,788 duplicated tweets were identified. After dropping these, 211,255 tweets remained for training.
- Step 3. Re-sampling dataset. As expected, the dataset had an imbalanced distribution, predominating positive tweets. This could affect the classifier training, thus a re-sampling process had to be performed, utilizing the numpy library and the function *numpy.random.choice* with `replace = True`. After re-sampling, 193,827 tweets were left.
- Step 4. The data was split into train and test datasets. Utilizing the class *sklearn.model_selection .train_test_split*, the dataset was divided where the train set contained the 70% and the test set the 30% of the original dataset. This was mainly done to prevent overfitting when training the model.
- Step 5. The based model was built, using a sklearn pipeline which combined the vectorization function and the SVM model. No hyper-parameters were tuned in this section.

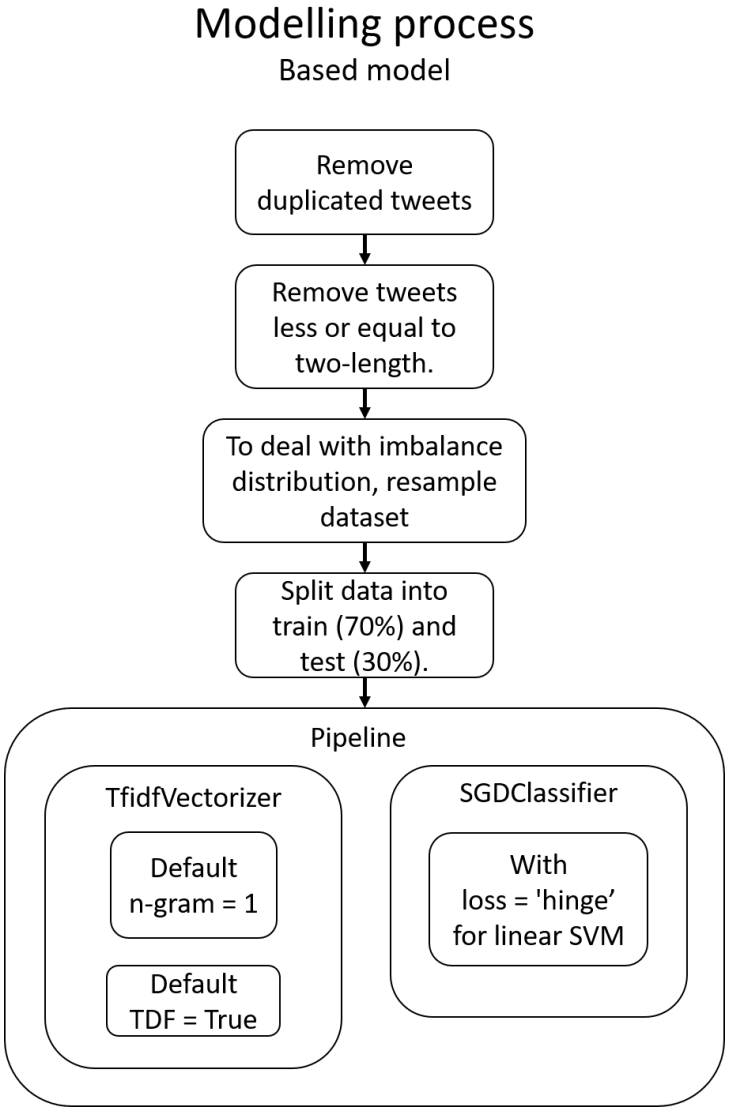


Figure 4.3: Data Collection process.

Listing 4.1 contains lines of code of the based model in Python. The whole Jupyter Notebook is available on the Github repository from *https://github.com/karla-cepeda/dissertation/code*. Table 4.2 sums up all core technology, platform, and programming language.

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import SGDClassifier

based_model_svm = Pipeline([
                            ('tfidf_vectorizer', TfidfVectorizer()),
                            ('svm', SGDClassifier())
                           ])
```

Listing 4.1: Based model libraries and pipeline.

Core technology
Stage: **Modeling process**
Programming Language: **Python**
IDE: **Jupyter Notebook**
Libraries: • **sklearn**. For pipeline, splitting dataset and create SVM model. • **matplitlib**. To plot graphs for evaluation. • **pandas** and **numpy**. For data manipulation. • **mysql_connection**. To connect to MySQL and select normalized tweets.

Table 4.2: Core technology for modeling stage.

Chapter 5

Parameter Tuning, Evaluation and Testing

Once the based model was designed and the training dataset was fitting into the pipeline, an evaluate process was carried out. The four metrics used were (sklearn 2021*b*, Rustam et al. 2021):

- **Accuracy**. It is the ratio of correct predictions which has a value between 0 and 1. The formula is shown in equation 5.1, where $\hat{y}_i$ is the predicted value of the i-th sample, y_i is the true value, n is the total samples, and $1(y)$ is the indicator function.

$$accuracy(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} 1(\hat{y}_i = y_i) \tag{5.1}$$

- **Precision**. This score is the fraction of the true positives over the total true positives in a specific class. The formula is shown in equation 5.2, where tp is true positives, and fp false positives. The range for this score is between 0 and 1.

$$precision = \frac{tp}{tp + fp} \tag{5.2}$$

- **Recall**. Indicates the ability of the classifier to find positive samples. This is the fraction of true positives within a class divided by the total observations in a specific class. The formula is shown in equation 5.3, where tp is true positives, and fn false negatives. Its value lies between 0 and 1.

$$precision = \frac{tp}{tp + fn} \tag{5.3}$$

- **F1-Score**. This metric is interpreted as a weighted average of the precision and recall scores. The formula is shown in equation 5.4:

$$F_1 = 2 * \frac{precision \times recall}{precision + recall} \tag{5.4}$$

Another technique used to evaluate the model was a Confusion Matrix, which summarizes correct and wrong observations within classes. A confusion matrix C has C_{ij} number of observations in a class i and predicted group j. The diagonal elements represent the true positive values predicted in a class, while the other elements are miscategorized observations. The higher the number, the better the model (sklearn 2021*f*).

After training the based model, the training accuracy score was 78.96%, and on the test, 78.03%. Table 5.1 shows a complete classification report using the class *classification_report*, where the average value in each metric is 78%. Figure 5.1 shows the confusion matrix on the test dataset, where there are many true positives on predicted values in the test set.

	Precision	**Recall**	**F1-Score**
Negative	0.75	0.84	0.78
Neutral	0.80	0.74	0.79
Positive	0.80	0.76	0.77

Table 5.1: Classification report from based model.

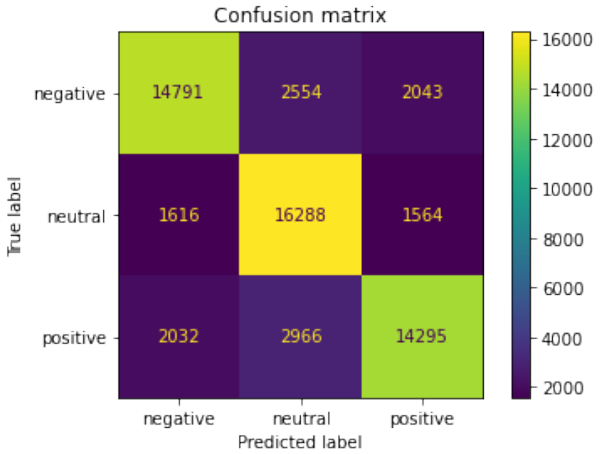


Figure 5.1: Data Collection process.

Although the results had an average accuracy score of 78%, hyper-parameters were tuned to improve this score and an inspection on over-fitting after finding the best parameters. The hyper-parameters tuned on the SVM algorithm were (sklearn 2021*g*):

- **penalty**. Regularization term. The values available are "l2" (also known as Ridge), "l1" (also known as Lasso), and "elasticnet". The default value is "l2". These techniques are used to reduced complexity in the model and prevent over-fitting. The difference between "l2" and "l1" is that "l1" will produce some values of the weights exactly zero whereas "l2" is near-zero (Pykes 2021, Bhattacharyya 2018).
- **alpha**. This is the regularization ratio. The default values are "0.0001". This hyper-parameter will be tuned with the values: 1e-6, $1e^{-5}$, $1e^{-4}$", $1e^{-3}$, $1e^{-2}$, 0.5, 0.75, and 1. Additionally, this parameter is used to compute the learning rate to decrease strength and upgrade the gradient of the loss function.
- **max_iter**. This is also known as "epochs". The maximum number of passes over the train set. The default value is "1000". This hyper-parameter will be tuned with the values: 1, 10, 50, 100, 500, 1000, 1500, 2000.

Additional parameters were included in the SVM model: `shuffle` to `True` (the data is shuffled after each epoch), `random_state` set to `1` (to control random generator), and `early_stop` to `True` (to terminate training when the validation score is not improving anymore). The `learning_rate` value by default is `optimal`, where the formula used is $eta = 1.0/(alpha(t + t0))$ (sklearn 2021*g*).

A grid search was executed to search for the best parameters on given specific values to avoid tweaking these hyper-parameters one by one on the classifier, using the class *sklearn.model_selection.GridSearchCV*. The parameters `max_iter` and `alpha` were tuned using the grid search, and it was used two times to assess the `penalty` parameter (also known as the regularization term) with the values *l2* and *l1*, separately[1]. By the number of values given to `alpha` and `max_iter`, the number of combinations was 64 for each penalty option l2 and l1. The scoring used to assess the results of the grid search was the *accuracy* score and set the parameter `return_train_score` to `True` to get train score values. Before fitting the grid search, the class *model_selection.StratifiedKFold* was used as a splitting strategy that provides a train and validation split on the dataset, preserving the percentage of samples within the classes. The number of folds set was four to split the data into 75% for training and 25% for validation (sklearn 2021*h*, Dekanovsky 2021). Listing 5.1 shows a code example of the grid search.

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.pipeline import Pipeline
from sklearn.linear_model import SGDClassifier
from sklearn.model_selection import GridSearchCV, StratifiedKFold

test_svm = Pipeline(
                      [
                          ('tfidf', TfidfVectorizer(ngram_range = (1,1),
    use_idf = True, lowercase = False)),
                          # Vectorizer to transform text into number
                          # ngram_range: Create a bag of word of individual
     tokens.
                          # lowercase: false, as text is already lowercase
    format.

                          ('svm', SGDClassifier(loss = 'hinge', shuffle =
    True, random_state = 1, early_stopping = True))
                          # Support Vector Machine
                          # loss = hinge is Suppor Vector Machine
                          # early_stopping: terminate training when
    validation
                          #   score is not improving
                          # shuffle: shuffled after each epoch
                          #   according to documentation, l1_ratio = 0 is
    l2,
                          #   default = 1.5
                      ]
```

[1]The grid search plot used to display results plots two parameters. Thus it was decided to assess the penalty parameter separately.

```
                    )

# Hyper-parameters to be tunned by GridSearchCV
parameters = {
                'svm__alpha': [1e-6, 1e-5, 1e-4, 1e-3, 1e-2, 0.5, 0.75,
    1],
                # Regularization rate
                'svm__max_iter': [1, 10, 50, 100, 500, 1000, 1500, 2000]
                # Epchos
             }

# Cross Validation
# This will split the data into train 75% and validation 25%
kfolds = StratifiedKFold(n_splits = 4, shuffle = True, random_state=1)

grid_svm = GridSearchCV(test_svm,
                        param_grid = parameters,
                        cv = kfolds,
                        scoring = 'accuracy',
                        n_jobs = -1, # Use all CPU
                        return_train_score = True
                        )
```

Listing 5.1: GridSearchCV for tuning hyper-parameters.

Different plots were used to assess the best hyper-parameters given by the grid search (sklearn 2021*c*):

- **Grid Search plot**. This compares the accuracy score of all possible combinations of the parameters. The library that will be used is *sklearn_evaluation*. There are different types of plots but the one that will be used is *plot.grid_search* and *matplotlib*.
- **Validation curve**. This is used to examine the alpha values and find out whether there is an over-fitting or under-fitting sign. If the validation errors are large and close to the training line, there is a sign of under-fitting, whereas if training and validation errors significantly differ it is a sign of over-fitting. The libraries used to plot it are *sklearn.model_selection.validation_curve* and *matplotlib*.
- **Learning curve**. This plot shows the different numbers of training samples and their training scores respectively. This helps on whether there is a benefit on adding more samples and whether suffers from a variance error or bias error. The libraries used are *sklearn.model_selection.learning_curve* and *matplotlib*.

After fitting the training set into the GridSearchCV with the `penalty` l2 a plot was created to compare and find the best score produced. Figure 5.2 shows the results, where the best parameters are `alpha = 1e-5` and `epochs = 10` with 0.816 of accuracy score. An alpha value different than $1e^{-5}$ would produce lower training and test accuracy score.

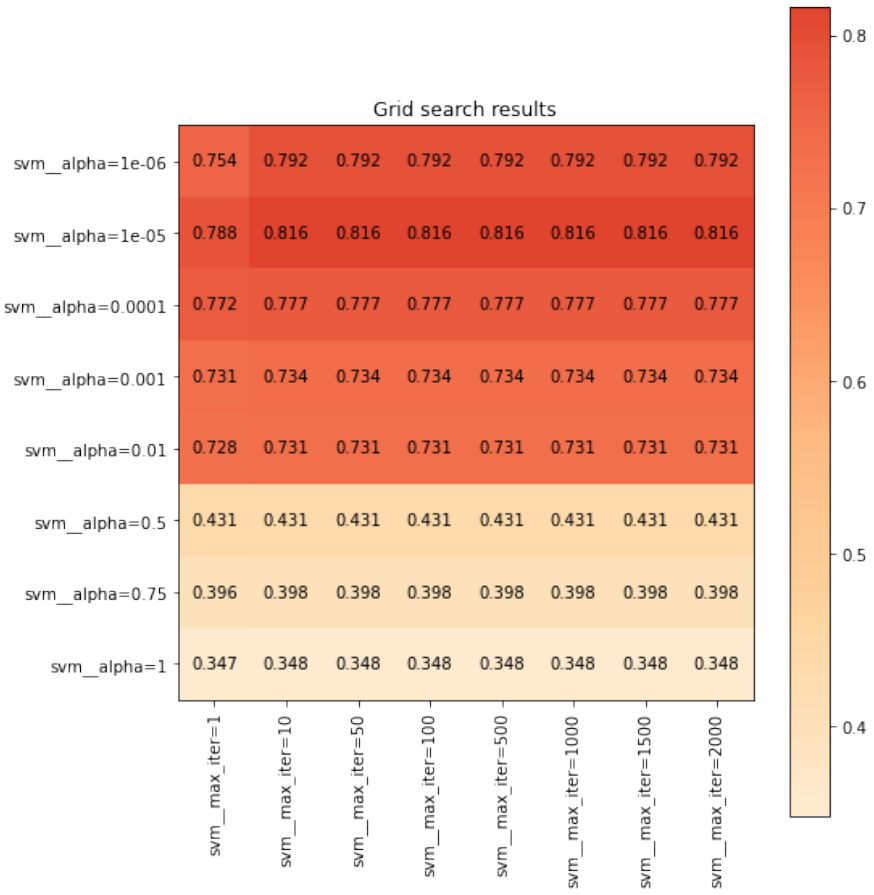


Figure 5.2: GridSearchCV results with l2 regularization.

Figure 5.3 shows the validation curves according to the alpha values on the grid search. The plot on the right shows alpha values and accuracy scores, and on the left presents the same data but with the four lowest alpha values. Alpha greater than 0.001 leaded to under-fitting as the train and validation errors were large and close to each other. On the other hand, when `alpha` was equal to $1e^{-6}$, there was a significant difference between training and validation accuracy scores, suggesting an over-fitting issue. An alpha equal to $1e^{-5}$ had good training and validation accuracy. When `alpha` was equal to $1e^{-4}$ and $1e^{-3}$, the accuracy score was lower. Therefore, the value for the learning rate taken was $1e^{-5}$.

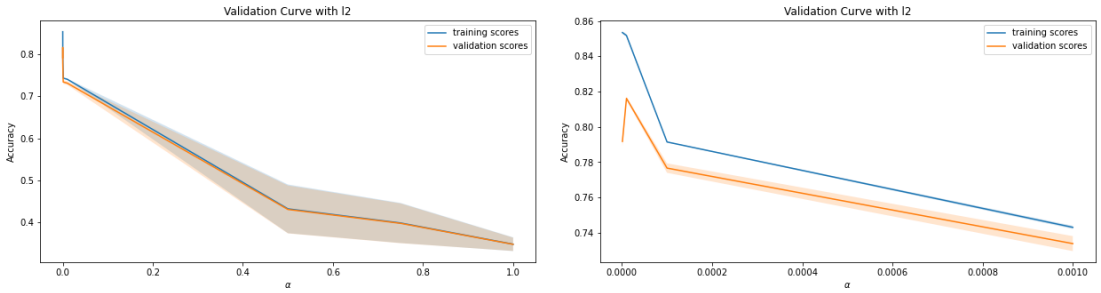


Figure 5.3: Validation curves with penalty l2.

Figure 5.4 shows the learning curve plot, where the accuracy score of the train set is 0.85 and for validation, 0.81. There was no sign of over-fitting as there was no significant fluctuation in the training and validation lines. Therefore, this was still a good alpha value that increased the accuracy score by 2% comparing to the based model.

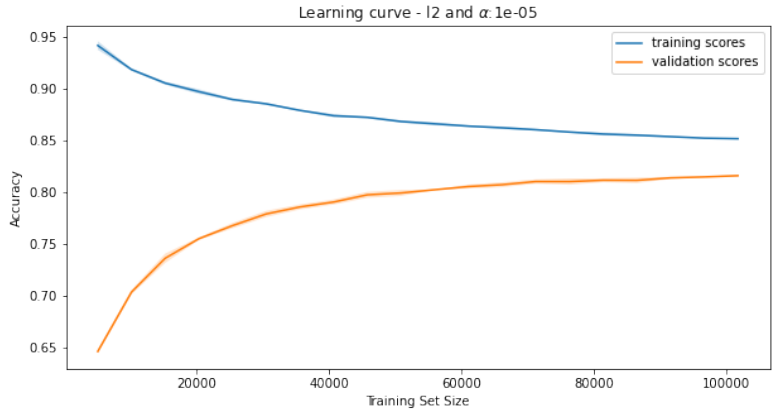


Figure 5.4: Learning curves, penalty l2 and alpha $1e^{-5}$.

Trying the `GridSearchCV` with a `penalty` of l1, the grid search plot is shown in Figure 5.2. The best parameter, in this case, are `alpha = 1e-5` and `epochs = 10` with 0.824 of accuracy score, a better accuracy score than `l2` regularizer. An alpha value lower or greater than $1e^{-5}$ produced lower accuracy scores.

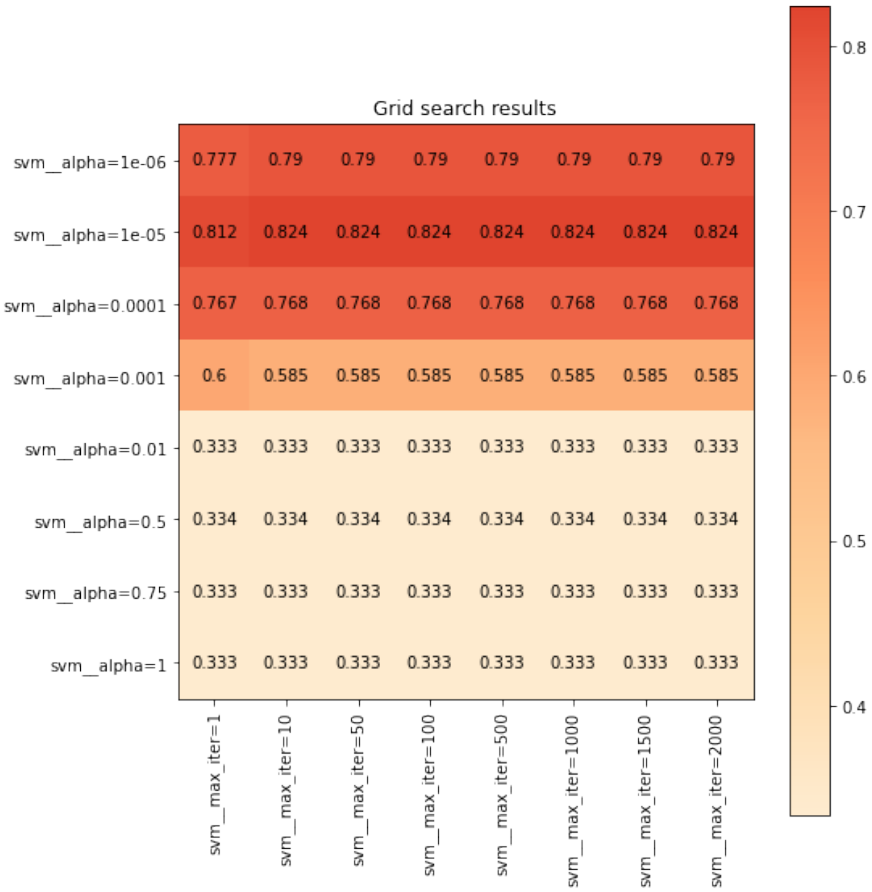


Figure 5.5: GridSearchCV results with l1 regularization.

Figure 5.6 shows the validation curves according to the `alpha` values on the `GridSearchCV`. The plot on the right shows alpha values and accuracy scores, whereas on the left presents the same data but the lowest four alpha values. Alpha greater than $1e^{-3}$ leaded to underfitting as the validation error was large and close to the training line. On the other hand, when alpha was equal to $1e^{-6}$, there was a significant difference between training and validation

accuracy scores, suggesting an over-fitting issue. For alpha $1e^{-4}$ and $1e^{-3}$, the accuracy score was low, whereas alpha equal to $1e^{-5}$ had a good training and validation accuracy and did not differ significantly. Hence, the value for the learning rate taken was $1e^{-5}$.

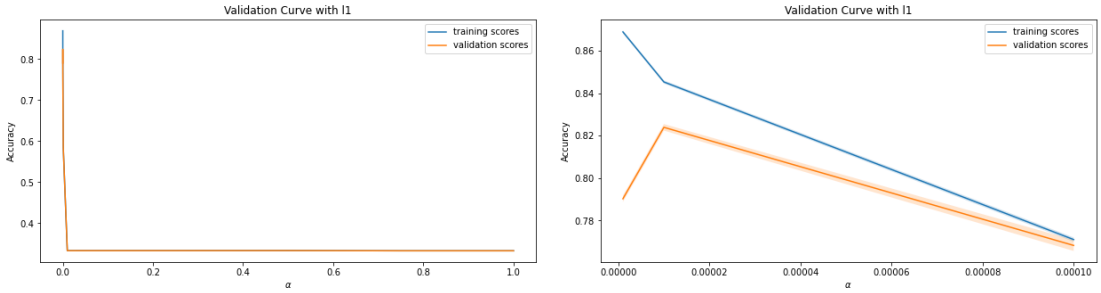


Figure 5.6: Validation curves with penalty l1.

Figure 5.7 shows the learning curve plot with alpha equals to $1e^{-5}$. The accuracy score of the train set is 0.84 and for the validation set is 0.82. There is no sign of over-fitting due to the shape of the training and validation lines. Therefore, this was still a good alpha value that increased the accuracy score by 3% comparing to the based model.

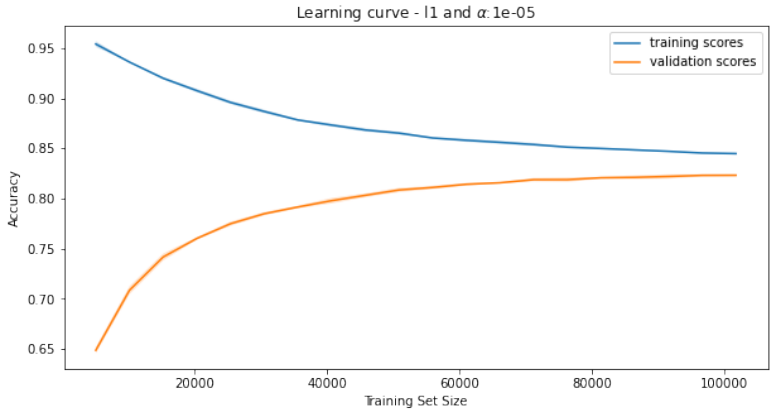


Figure 5.7: Learning curves, penalty l1 and alpha $1e^{-5}$.

Comparing the best parameters from the `GridSearchCV` executed for `l2` and `l1`, the one that has the best accuracy score is `l1` with alpha $1e^{-5}$ and 10 epochs. Table 5.2 shows a summary of the values tuned using `GridSearchCV`. Therefore, the final model had the following hyper-parameter values: `penalty = l1`, `alpha = 1e-5`, and `epochs = 10`.

GridSearchCV best parameters			
Penalty	**Alpha**	**Epochs**	**Accuracy**
l2	$1e^{-5}$	10	0.81
l1	$1e^{-5}$	10	**0.82**

Table 5.2: Comparison of the best parameters within l2 and l1.

Finally, the function *sklearn.model_selection.cross_val_score* and the class *sklearn.model_selection.RepeatedStratifiedKFold* were used to generate four folds that were repeated 15 times, producing different splits in each repetition (sklearn 2021*a*). After getting the accuracy scores from each repetition, the graph in Figure 5.8 shows 15 box-plots and the mean (with an orange line) and median (with a green triangle) in each box-plot. As the media and the median values roughly coincide, it is suggested a reasonable symmetric distribution and the mean

may capture the central tendency well. Thus, the test harness and the model appeared to be a **good choice** (Brownlee 2020).

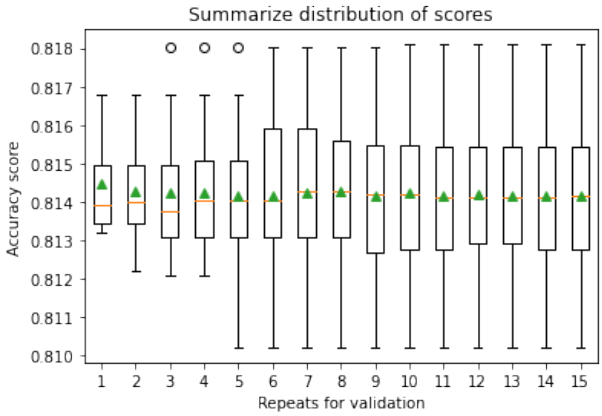


Figure 5.8: Distribution of repeated cross-validations.

For the final model, the accuracy score on the training dataset was 84%, and for the test dataset was 82%. By tuning the hyper-parameters, the model improved compared to the based model. A complete classification report is shown in Table 5.3, where the average value in each metric is 83%. Figure 5.1 shows the confusion matrix on the test dataset, where there is a good amount of true positives on predicted values in the test set. Figure 5.1 shows the confusion matrix on the test dataset with the final model. There is an improvement in the number of true positives of "negative" and "positive" classes on predicted test values comparing to Figure 5.1.

	Precision	**Recall**	**F1-Score**
Negative	0.82	0.84	0.83
Neutral	0.86	0.81	0.84
Positive	0.81	0.83	0.82

Table 5.3: Classification report from final model on test dataset.

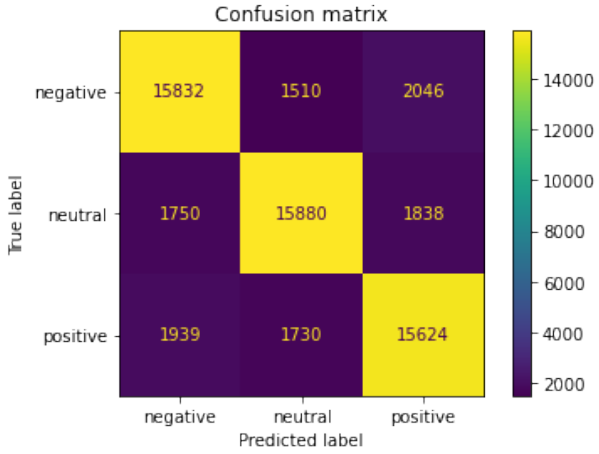


Figure 5.9: Confusion matrix from final model on test dataset.

Chapter 6

Results

After building the final model for sentiment classification, the Irish tweets were classified using this model and stored the results in the DB. In this section, one-word tweets with a length of less than three were considered empty tweets. Moreover, these empty tweets were not classified as *neutral* due to the lack of characters. The collection process extracted 155,950 tweets from the API. After removing all tweets tagged as empty, there were 141,886 tweets for the Sentiment Analysis. All tweets are considered for a "global sentiment" analysis level. For a "recent sentiment" level, tweets created from the last week are taken for the analysis[1], whereas, for a "last sentiment" level, tweets created at the 13th of August 2021 are considered for the analysis (the last date recorded in the DB for this report). Line plots and pie charts used for the analysis were created with the *plotly* library on a Jupyter Notebook file. Eventually, this functionality of the plots was implemented on a dashboard using the *dash* library.

Figure 6.1 shows the daily sentiment over time from March 2021 and August 2021, where, in general, the predominated sentiment is *negative*. From the last week, *negative* tweets predominate. The sentiment is still *negative* on the 13th of August 2021 (the last date recorded in the DB).

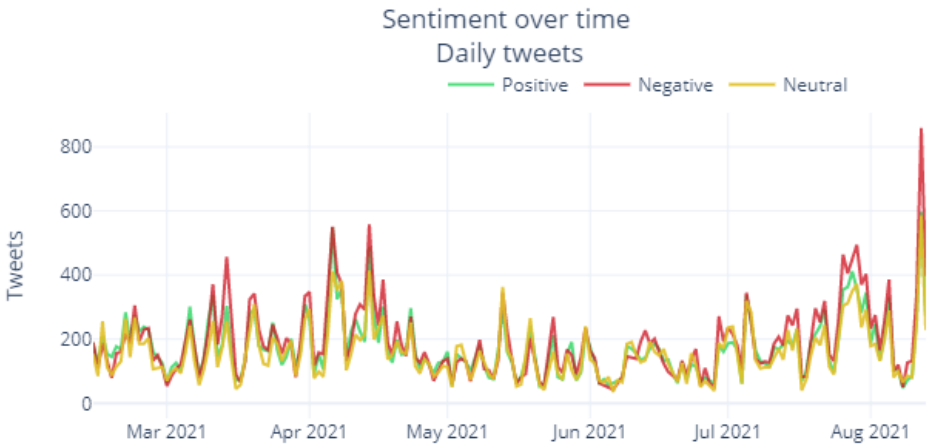


Figure 6.1: Daily sentiment in Ireland.

[1]Monday to Sunday, for this report corresponds two the second week of August (from 2021-08-09 to 2021-08-15). However, the data is still updating due to the daily collector of tweets.

Figure 6.2 shows the weekly sentiment over time from January 2020 to August 2021. In general, the green line (positive sentiment) surpasses the red (negative sentiment) and yellow (neutral sentiment) lines before February 2021, assuming that the predominated feeling in this period is *positive*. However, the *negative* sentiment starting predominating after February 2021. This shift in the sentiment may occur due to the starting of the vaccine program in Ireland and the increasing number of news related to the vaccines tweeted by Irish usernames (see Figures 3.13 and 6.4) (during the collection process, conversations had a more significant proportion of tweets than another batch category, see Section 3.4 Description of the data). Figure 6.3 shows a monthly distribution of the sentiment, where it is more evident how the *negative* feeling took over after February 2021. On the right side of the figure, the pie chart shows that 34.4% represents the positive tweets, 35.8% of tweets are classified as *negative*, and neutral opinions are 29.8%. Notably, the difference between positive and negative distribution is insignificant.

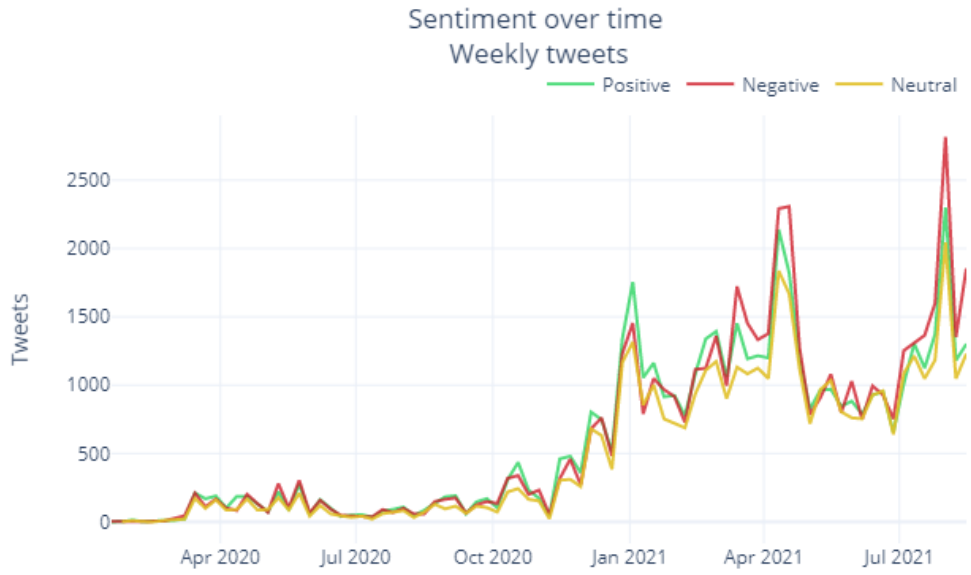


Figure 6.2: Monthly sentiment in Ireland.

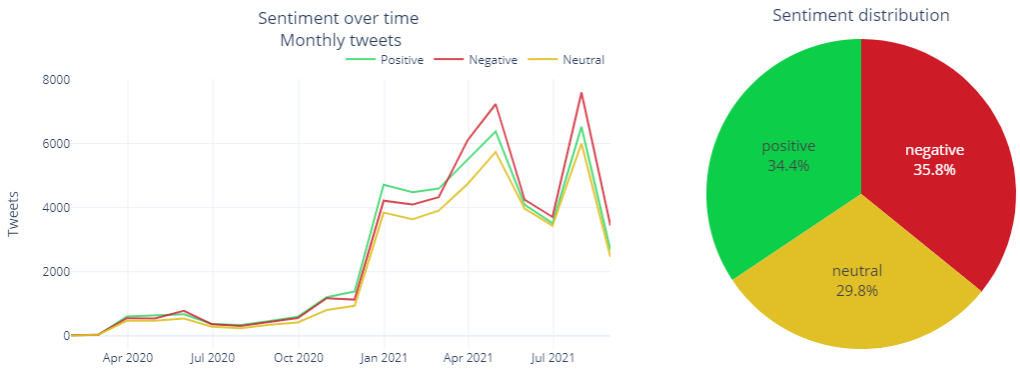


Figure 6.3: Monthly sentiment in Ireland.

The following figures are shown weekly due to the fluctuation in the daily number of tweets. Figure 6.4 shows the weekly sentiment over time on Irish media usernames and a pie chart to summarize this distribution. These types of users have tweeted 3,779 times, where the *neutral* feeling predominated. It seems there is a lack of sentiment on the tweets that Irish media usernames have tweeted. However, this could be a result of the nature of the new. It is interesting to see how the *positive* and *negative* tweets from Irish media usernames have similar proportions, 27.2% and 23%, respectively, whereas the *neutral* is almost half the number of all tweets. At a "recent sentiment" level, the predominated sentiment is *neutral*, whereas, at a "last sentiment" level, it is still *neutral*.

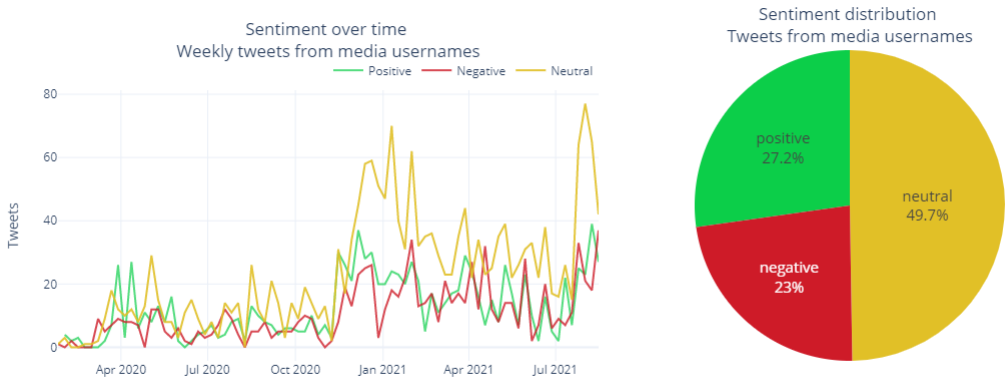


Figure 6.4: Irish media usernames' sentiment.

Figure 6.5 shows the sentiment on Irish government (such as departments and politicians) over time. Although these types of usernames tweeted 1,109 times, by looking at the pie chart, *positive* and *neutral* tweets have been posted more frequently with a proportion of 49.1% and 39.1%, respectively, whereas tweets with negative sentiment have a distribution of almost 18%. In a "recent sentiment" level, the predominated sentiment is *positive*, whereas, on the last date, it is still *positive*. Comparing with the distribution of tweets posted by media usernames, this suggests that the government usernames have been careful on tweeting, tending to post *positive* or *neutral* news/messages related to the Covid-19 vaccines.

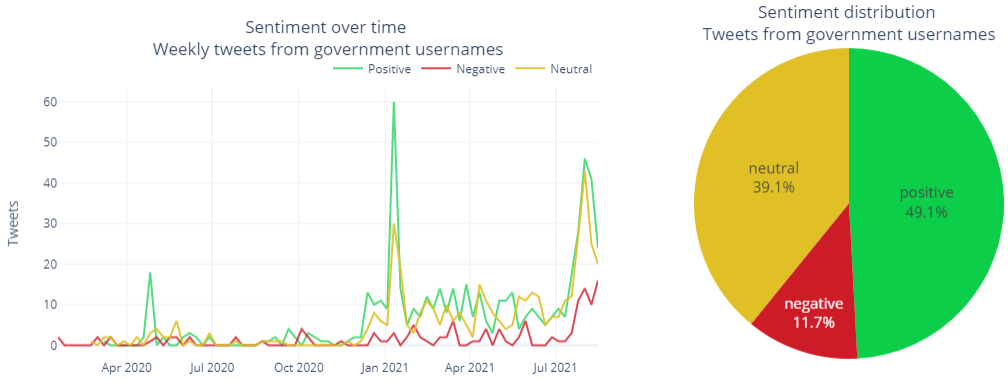


Figure 6.5: Irish government and department usernames' sentiment.

Figure 6.6 shows Irish political party usernames and the sentiment over time. These have tweeted 339 times, where 53.1% are *positive*, 29.8% are neutral, and 17.1% are negative. At a "recent sentiment" level, the predominated sentiment is *neutral* There is not enough data for a "latest sentiment" level since there is one positive tweet and one negative tweet. Notably, these usernames have a similar sentiment proportion as government usernames shown in Figure 6.5.

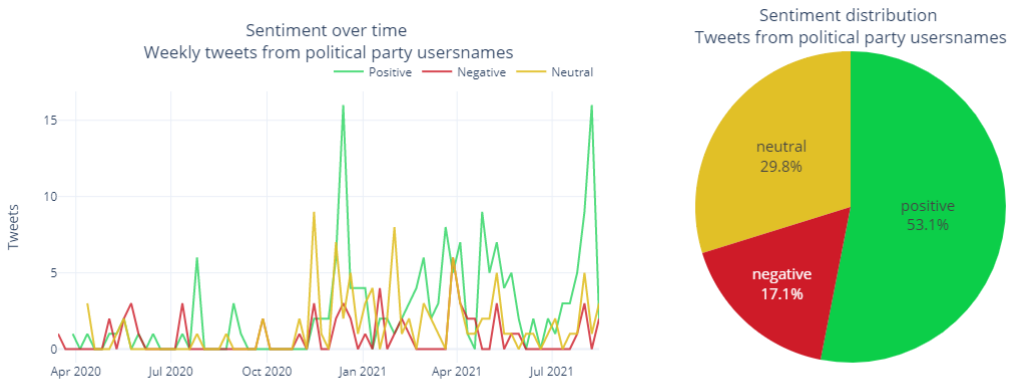


Figure 6.6: Irish political party usernames' sentiment.

Figure 6.7 shows Irish health department usernames. These types of users have tweeted 70 times, where 48.6% are *positive*, 34.3% are neutral, and 17.1% are negative. At a "recent sentiment" level, the predominated sentiment is negative. There is no data for a "last sentiment" level. Notably, these usernames have a similar sentiment proportion as usernames shown in Figure 6.5 and 6.6.

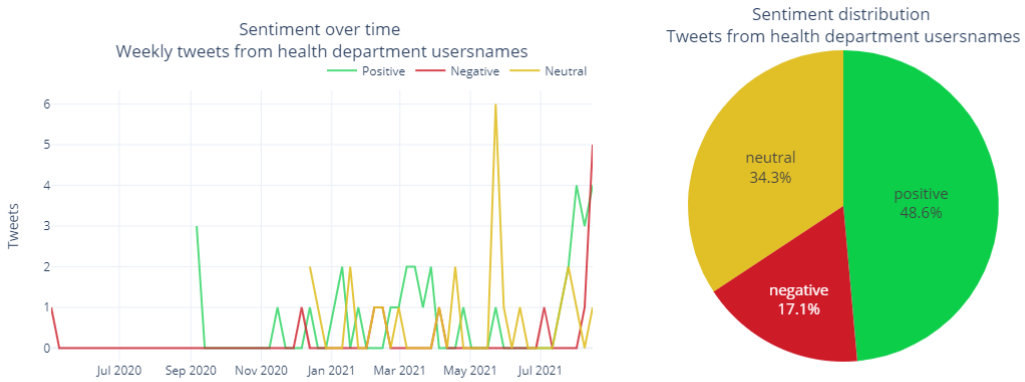


Figure 6.7: Irish health department usernames' sentiment.

The following figures show the sentiment on tweets that have mentioned *Vaxzevria*, *Comirnaty*, *Moderna*, *Janssen*, and related pharmaceutical names over time. There were two options to select these tweets: taking tweets containing the name of one vaccine (despite another vaccine is considered) or selecting tweets that mentioned just one vaccine. It seems that the overall distribution of the feeling has a similar proportion when comparing these two approaches. Therefore, the first approach was taken to plot the sentiment and the proportion over time. Although, these results should be taken carefully as the amount of data is insufficient to suggest what vaccine is the most "accepted" in Ireland.

Figure 6.8 shows two plots: on the left side is the weekly sentiment from January 2020 to August 2021, whereas on the right side is the distribution of tweets and sentiment. For *Vaxzevria*/*Astrasenecan* vaccine, there are 7,354 tweets in total, where 231 are from batch usernames (i.e., media, government, political party, and health usernames), and 7,081 are replies or from "other" usernames. It seems that the "global sentiment" from this vaccine/pharmaceutical is *negative*. However, there is no significant difference when comparing to the number of positive tweets. The pie chart shows that 37.2% of tweets are *negative*, whereas 33.3% are positive and 29.5% are neutral. For a "recent sentiment" level, the *negative* tweets are still predominant. There is not enough data for a "latest sentiment" level analysis since there are just two positive tweets and two negative tweets. The number of tweets is ***too low*** to take these results as general sentiment in Ireland.

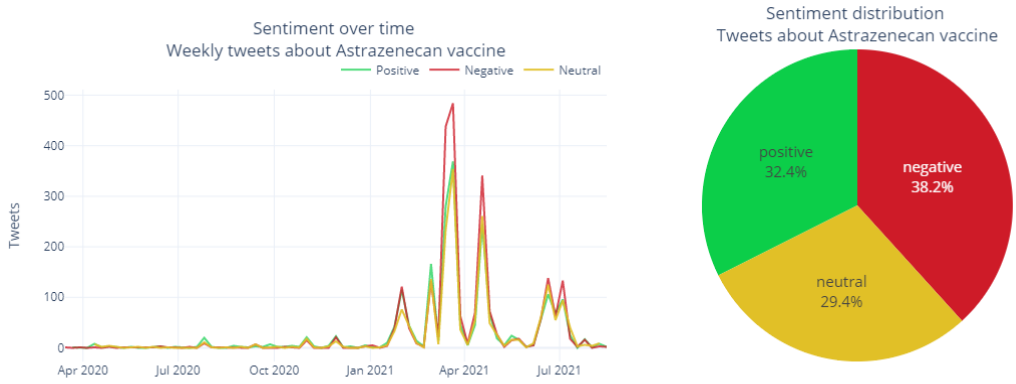


Figure 6.8: Sentiment on Vaxzevria/Astrazenecan vaccine.

Figure 6.9 shows the weekly sentiment on the *Comirnaty*/*Pfizer* vaccine from January 2020 to August 2020. In general, 4,671 tweets were collected related to this vaccine, where 267 are tweets from batch usernames (i.e., media, government, political party, and health) and 4,404 are replies or from "other" usernames. It seems there is a predominance of *positive* feelings. The pie chart on the right side shows that 38.2% of the tweets are *positive*, whereas the negative tweets have a percentage of 29.8 and neutral tweets 32%. At a "recent sentiment" level, the *positive* feeling is still predominant. However, there is not enough data for a "latest sentiment" level since there are three positive tweets and one negative tweet. The amount of tweets is ***too low*** to take these results as general sentiment in Ireland.

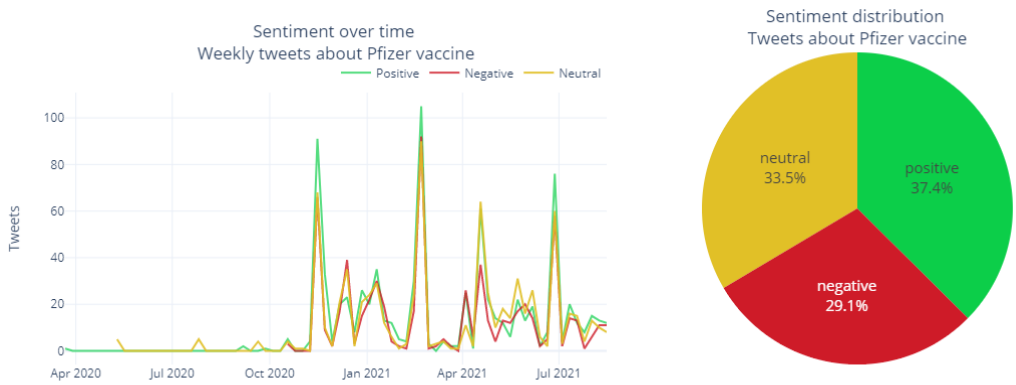


Figure 6.9: Sentiment on Comirnaty/Pfizer vaccine.

Figure 6.10 shows the weekly sentiment on the *Moderna* vaccine from January 2020 to August 2021. The initial collector gathered 1,361 tweets where 90 tweets are from batch usernames (i.e., media, government, political party, and health), and 1,271 are replies or from "other" usernames. Like the *Comirnaty*/*Pfizer* vaccine, this has a predominance of *positive* feelings. The pie chart on the right side shows that 41.4% of the tweets are *positive*, whereas the negative tweets have a percentage of 27.4 and neutral tweets 31.2%. At a "recent sentiment" level, *positive* tweets are predominated. There is not enough data for a "latest sentiment" since there are two positive tweets and one negative tweet. However, the amount of tweets is ***too low*** to take these results as general sentiment in Ireland.

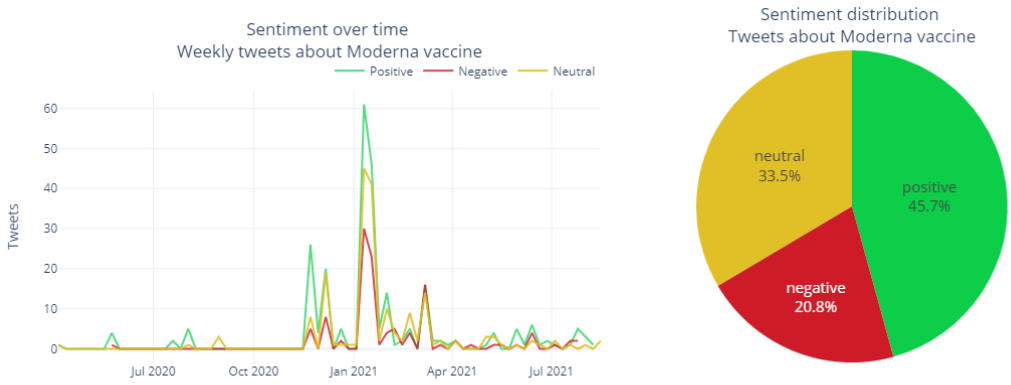


Figure 6.10: Sentiment on Moderna vaccine.

Lastly, Figure 6.11 shows the weekly sentiment on the *Janssen*/*JJ* vaccine. The initial collector gathered 3,644 tweets related to this vaccine, where 506 are from batch usernames (i.e., media, government, political party, and health), and 3,138 are replies or from "other" usernames. It seems that *negative* feelings are predominant. The pie chart shows that 34.7% of the tweets are *negative*, whereas the positive tweets have a percentage of 31.8 and neutral tweets 33.5%. At a "recent sentiment" level, *positive* tweets are predominated. There is not enough data for a "latest sentiment" level since there is one positive tweet and one negative tweet. However, the amount of tweets is too low to take this result as general sentiment in Ireland.

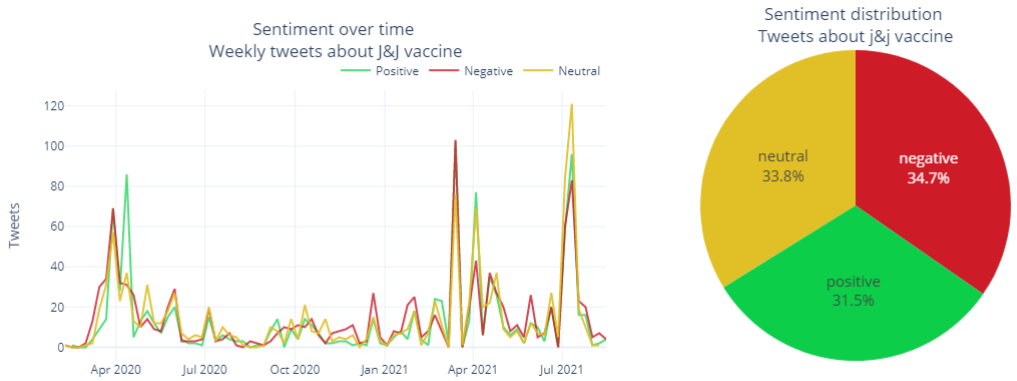


Figure 6.11: Sentiment on Janssen/J&J vaccine.

Chapter 7

Conclusion

A Sentiment Analysis was carried out to find and analyze the opinion on the Covid-19 vaccines in Ireland. The methodology implemented for this project was based on the CRISP-DM strategy. Before starting with collecting tweets, access to the Research Academic product track was granted by Twitter to call the API. Batches were designed to build specific queries to call the endpoint "Full-archive search" and download tweets containing the keywords "covid," "vaccine," and other synonyms or related words. The initial collection was designed on Python. This code generated two datasets: global tweets (excluding Ireland) for labeling and modeling purposes and Irish tweets to analyze the sentiment. After this, a cleaning process was performed by removing no significant parts such as mentions, punctuation marks, and URLs. Subsequently, these cleaned tweets were normalized by removing stop words and the lemmatization process. Then, an exploration of the datasets was carried out using the *seaborn* and *matplotlib* libraries in Python. One-word tweets with length two or less were treated as empty tweets and digitally removed from the DB because of the lack of length and sentiment. Before the modeling stage, a labeling process was required to train the Machine Learning (ML) algorithm, where the lexicon-based tool Valence Aware Dictionary for sEntiment Reasoning (VADER) was the most accessible, affordable, and outperformed option. Different ML algorithms were discussed, where a Support Vector Machine (SVM) was chosen due to its outstanding performance reported by different academic papers. Later, a based model was designed using a pipeline that combines the SVM and the TFIDF vectorizer. After tuning the hyper-parameters *alpha* and *epochs* on *l1* and *l2*, the model achieved an accuracy of 82%. Even though the opinion on the Covid-19 vaccines used to be positive before February 2021, a negative feeling was predominated in recent days. However, there was no significant difference comparing to positive tweets. From the 1st of January 2020 to the 13th of August 2021, a negative feeling was found at a global sentiment level. There was still a negative opinion on the vaccines in a recent sentiment level (second week of August 2021), whereas, in the latest sentiment level (the 13th of August 2021), the feeling was still negative. By inspecting the sentiment by type of usernames within the batches designed in the collection process, the media usernames have tweeted the most, with the most significant proportion of tweets classified as neutral. In contrast, tweets from government, political parties, and health usernames had a large proportion of tweets classified as positive. This suggested that most of these usernames were careful on what to share on social media. When analyzing the data by type of vaccine, the *Pfizer* and

Moderna vaccines showed a positive feeling and the *Astrazenecan* and *Johnson & Johnson* vaccines a negative sentiment. However, due to the insufficient data collected by vaccine's names, it was not possible to suggest what was the most "popular" vaccine in Ireland. Additionally, four courses were taken on DataCamp from May to June 2021 to understand the topic and conclude this project. To summarize, the contribution of this project is that it presented a discussion on the labeling and modeling process for Sentiment Analysis and examines the Irish tweets after classifying them as positive, negative, and neutral using a Support Vector Machine algorithm.

For future work, it will be interesting to research options to collect data from other social media like Facebook, Reddit, and YouTube to increase information and cover more data sources. Furthermore, it would be interesting to investigate unsupervised learning approaches to classify the sentiment of social media data. Additionally, an analysis on an emotion-grained would have been interesting to perform and analyze on this dataset. The data collected and labeled will be fascinating to understand if there is a correlation between the number of tweets, the sentiment, the number of people vaccinated, the new cases, and the number of deaths. Moreover, this will be useful for time series analysis, e.g., to forecast the number of people that would take a vaccine (It may require additional information such as the number of vaccinated people and the number of new Covid-19 cases from previous days). This will be an exciting project for the future to forecast the life cycle of each vaccine and inventory management to avoid expiration issues. Additionally, it will be interesting to apply the same methodology in a different location, such as Mexico. However, it will be needed to look into NPL libraries in Python that support the Spanish language.

Chapter 8

Deployment

A dashboard, a daily collector, a dates collector, and a remote DB were designed to share and display the results presented in this document and keep track of daily tweets. Figure 8.1 shows the structure of the remote DB named Twitter[1]. The following list describes each table:

- **tweet**. This table is used to stored relevant information related to the tweets for Sentiment Analysis. Due to the terms of use stated by Twitter API, the data migrated to this table are tweet_id, created_at, conversation_id, and author (this column contains batch usernames defined in the collection process, and other users were tagged as "Others"). The researcher created the columns keywords, keywords_pharma, label_id, and label. Therefore, there is no problem migrating these into the remote DB.
- **date**. This table stores dates/events from the *"Timeline of the Covid-19 pandemic in Ireland"* on Wikipedia[2]. This data was used as a source to collect significant dates related to Covid-19 and vaccines daily.
- **reference**. This table stores the references from each event within the table *date*. These events came from news and articles from broadcasters and newspapers such as RTE and The Irish Times.
- **date_reference**. This table stores the relationship between date and reference as there are many dates/events that the text has been taken from different sources.
- **tweet_date**. This table contains the relation between tweet and date based on the date. The data is used as a bridge to create a relationship since a tweet could have more than one event (i.e., date within table date) related.

The daily collector was coded in Python and was easy to implement since the scripts from the initial collection include a parameter that controls whether the process is for the "initial" or "daily" collection process. Additionally, the file used to design the batches allows setting default parameters for the daily collector. After the collection has been completed, the following steps are performed:

[1] The researcher has a personal hosting services provider, which has been used in another project in the second semester, there were no problems.

[2] Available from: https://en.wikipedia.org/wiki/Timeline$_{o}f_{t}he_{C}OVID-19_{p}andemic_{i}n_{t}he_{R}epublic_{o}f_{I}reland$

- Cleaning and normalization of the data.
- Find words less or equal to two-character length and change value active = 0.
- Label daily tweets with model created.

Once the tweets are stored in the DB, another process is triggered to migrate specific data into the remote DB. Then, a scraping process is triggered to extract information on the website Wikipedia to extract dates related to Covid-19 and vaccines in Ireland. This process accesses a timeline of the pandemic, inspects every link from this list, and extracts all dates, events, and references to save this into the remote DB. The *schedule* library was imported and set to 23:50 to trigger the daily process on the current date.

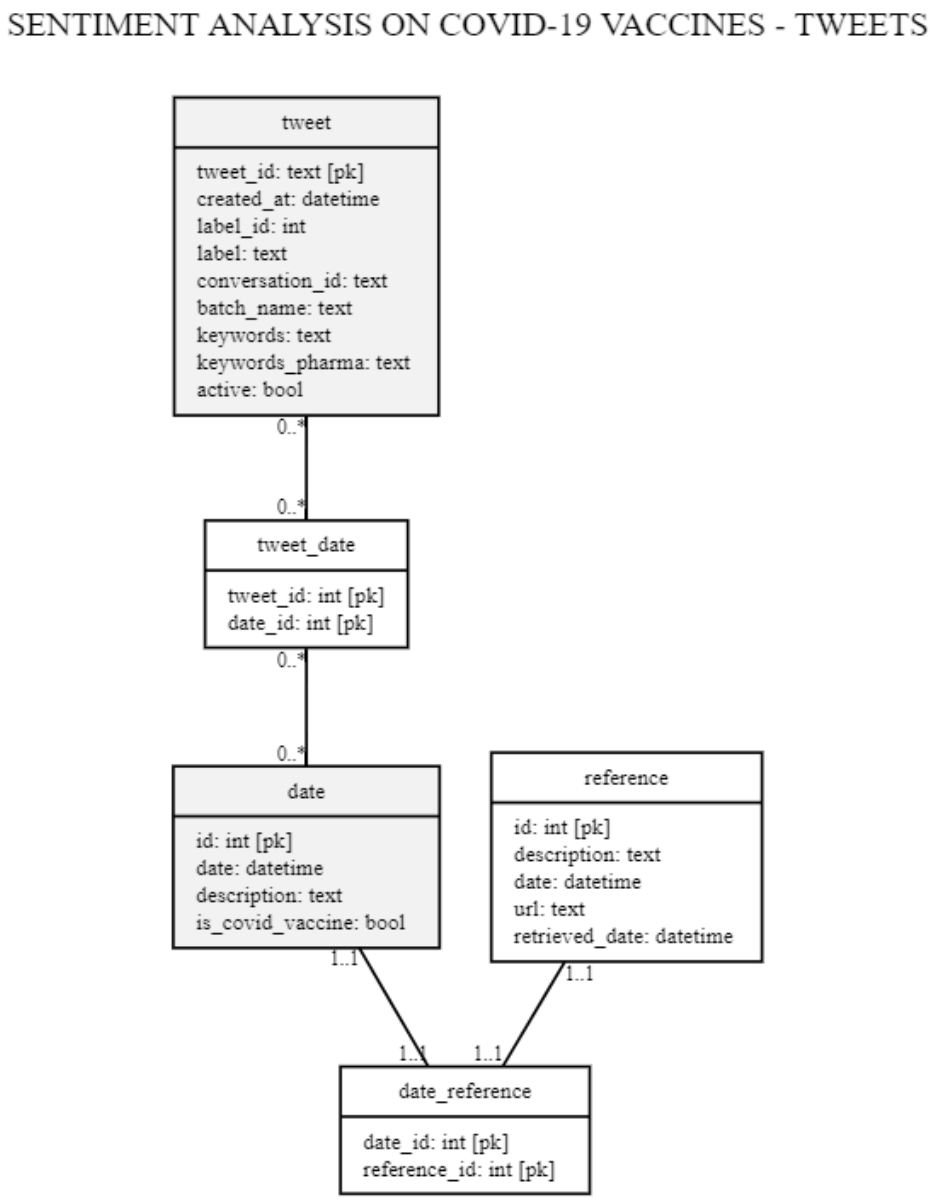


Figure 8.1: Remote database structure

The dashboard was initially designed in Power BI as it is an easy tool to use and has the functionality to publish and share the dashboard with others. Figure 8.2 shows the final dashboard on Power BI Desktop. After deploying the dashboard to the online service, an error in one of the sections was displayed due to the use of Python code blocks. The error says, "*Python visuals are a Power BI Pro feature. Only users with a Power BI Pro license can create, view, or interact with Python visuals.*" Therefore, this option was discarded; however, still available in *https://app.powerbi.com/groups/me/reports/1f05579e-23be-4f01-8495-db36cde2bd8c?ctid=abcc9a4b-cf04-4ca3-9d83-3465e3f43f61&pbi_source=linkShare*.

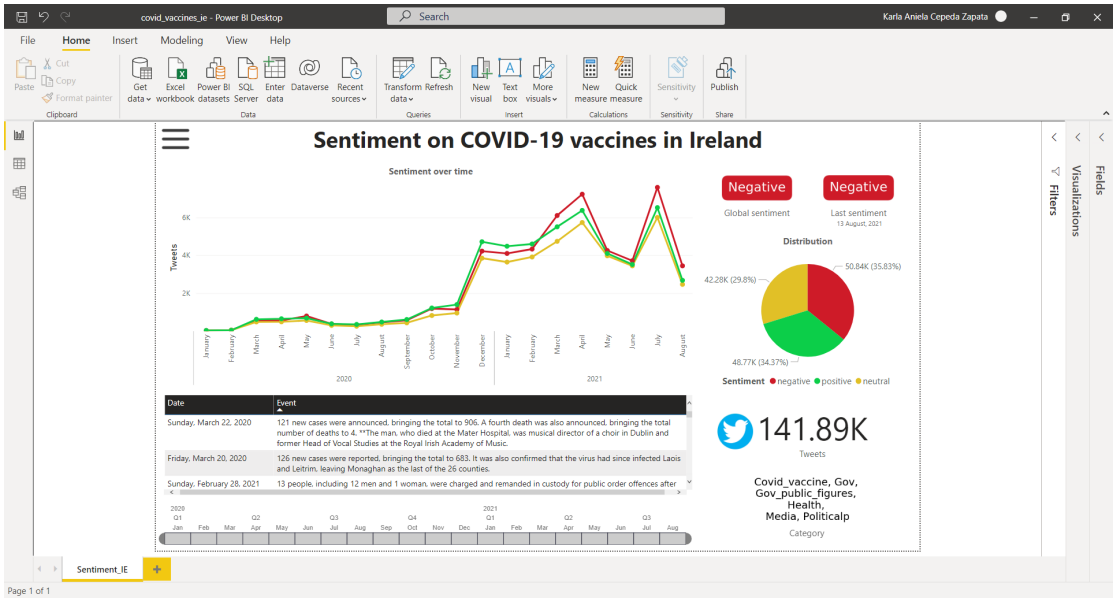


Figure 8.2: Dashboard built on Power BI

A second dashboard was designed on Python, using the library dash for the structure of the website and the library "plotly" to plot a line chart and pie chart. Two pages were designed: the home page, which describes the project and the data, and the dashboard page, which shows the sentiment and proportion over time. The dashboard includes filters to control the data to be displayed on the dashboard. The dashboard is aimed at the public domain, using a palette of green, red, and yellow colors to represent the positive, negative, and neutral opinions, respectively. This dashboard was deployed on Heroku, a cloud platform that supports many services in different programming languages. Figure 8.3 shows the final dashboard already deployed on Heroku. This is available from *https://sentimentanalysis-c19v-ie.herokuapp.com/*. Figure 8.4 shows the architecture of the final app.

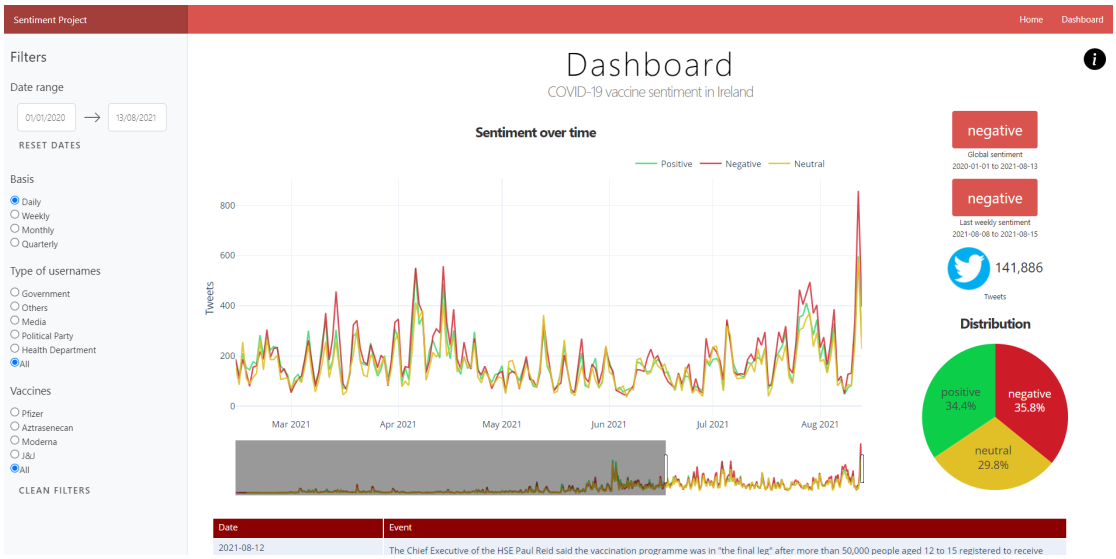


Figure 8.3: Dashboard print screen.

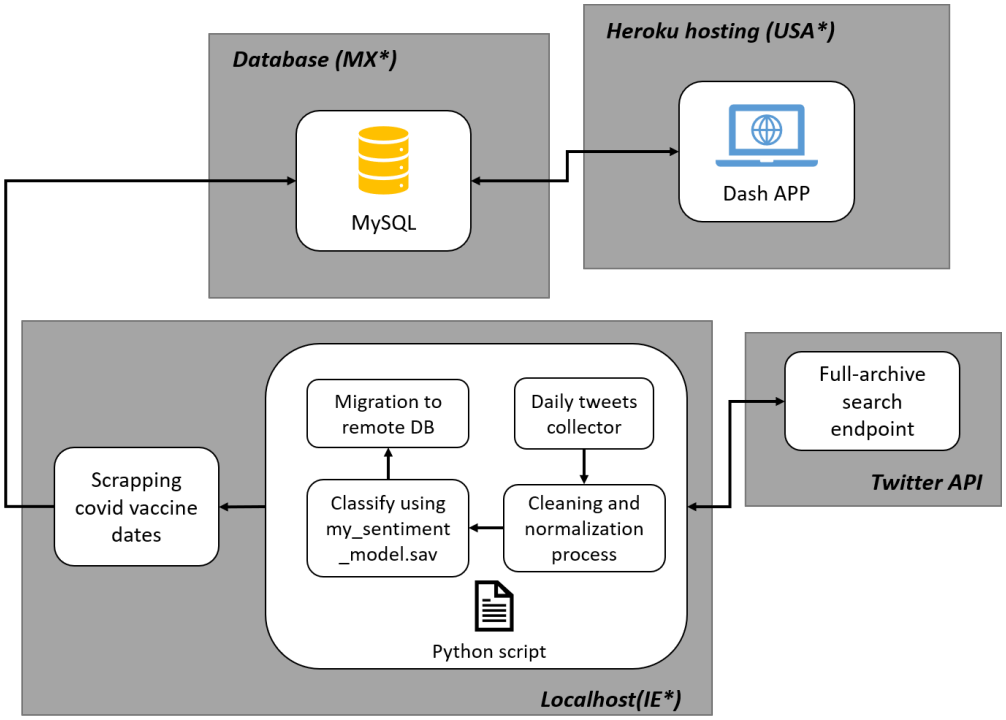


Figure 8.4: Structure of the dashboard.

Appendix A

A.1 Twitter API code example in Python

Description: Code example from Twitter API.
Authors: virgildotcodes and jdemos GitHub users.
Provided by: Twitter, Inc.
Available from:
https://github.com/twitterdev/Twitter-API-v2-sample-code/blob/master/Recent-Search/recent_search.py

```
import requests
import os
import json

# To set your environment variables in your terminal run the following line:
# export 'BEARER_TOKEN'='<your_bearer_token>'


def auth():
    return os.environ.get("BEARER_TOKEN")


def create_url():
    query = "from:twitterdev -is:retweet"
    # Tweet fields are adjustable.
    # Options include:
    # attachments, author_id, context_annotations,
    # conversation_id, created_at, entities, geo, id,
    # in_reply_to_user_id, lang, non_public_metrics, organic_metrics,
    # possibly_sensitive, promoted_metrics, public_metrics,
    referenced_tweets,
    # source, text, and withheld
    tweet_fields = "tweet.fields=author_id"
    url = "https://api.twitter.com/2/tweets/search/recent?query={}&{}".
    format(
        query, tweet_fields
    )
    return url


def create_headers(bearer_token):
    headers = {"Authorization": "Bearer {}".format(bearer_token)}
    return headers


def connect_to_endpoint(url, headers):
    response = requests.request("GET", url, headers=headers)
    print(response.status_code)
    if response.status_code != 200:
        raise Exception(response.status_code, response.text)
    return response.json()

```

```
41
42 def main():
43     bearer_token = auth()
44     url = create_url()
45     headers = create_headers(bearer_token)
46     json_response = connect_to_endpoint(url, headers)
47     print(json.dumps(json_response, indent=4, sort_keys=True))
48
49
50 if __name__ == "__main__":
51     main()
```

A.2 First code created to collect tweets.

Description: Fist version code created to collect Twitter data. This code is subjected to modifications.
Authors: Karla Cepeda.
Available from: https://github.com/karla-cepeda/Dissertation

```
1
2
3 import requests
4 import yaml
5 import urllib.parse
6 import os
7 import pandas as pd
8 from datetime import datetime, date, timedelta
9 import calendar
10 from dateutil.relativedelta import relativedelta
11 import time
12 import json
13
14 os.chdir(r'E:\Karla\IRELAND v2\DKIT\2nd Semester\Dissertation\Code\
       search_tweets')
15
16 def create_twitter_url(start_date, end_date):
17
18     words = '(vaccine OR vaccines OR vaccinated OR vaccination OR dose OR
        doses OR injection OR pfizer OR moderna OR NIAID OR astra OR
       astrazeneca OR oxford OR BioNTech OR "mRNA-1273" OR "johnson & johnson
       " OR "j&j" OR #vaccine OR #vaccines OR #vaccinated OR #AstraZeneca OR
       #pfizer OR #Moderna) (covid OR corona OR coronavirus OR covid19 OR "
       covid-19" OR virus OR "sars-cov-2" OR "sars cov 2" OR nCoV OR #covid19
        OR #covid)'
19     place = '(#ireland OR #dublin OR #galway OR #cork OR place:dublin OR
       place:cork OR place:galway OR place_country:IE)'
20     language = 'lang:en'
21     config =  "-is:nullcast"
22     q = "{} {} {} {}".format(words, place, language, config)
23
24     max_results = 500
25     mrf = "max_results={}".format(max_results)
26
27     start_date = "start_time={}".format(start_date)
28     end_date = "end_time={}".format(end_date)
29
30     expansions = "expansions=referenced_tweets.id,author_id,geo.place_id&
       tweet.fields=attachments,author_id,context_annotations,conversation_id
       ,created_at,entities,geo,id,in_reply_to_user_id,lang,public_metrics,
       possibly_sensitive,referenced_tweets,reply_settings,source,text,
       withheld&user.fields=created_at,description,entities,id,location,name,
       pinned_tweet_id,profile_image_url,protected,public_metrics,url,
       username,verified,withheld&place.fields=country,id,name"
31
```

```
    url = "https://api.twitter.com/2/tweets/search/all?query
    ={}&{}&{}&{}&{}".format(urllib.parse.quote(q), mrf, start_date,
    end_date, expansions)

    return url


def process_yaml():
    with open("config.yaml") as file:
        return yaml.safe_load(file)

def create_bearer_token(data):
    return data["search_tweets_api"]["bearer_token"]

def twitter_auth_and_connect(bearer_token, url):
    headers = {"Authorization": "Bearer {}".format(bearer_token)}
    response = requests.request("GET", url, headers=headers)
    return response.json()

def save_tweets(data, filename, foldername):

    #header = True
    #tweets = pd.DataFrame(data['data'], columns=data['data'][0].keys())

    folders = foldername.split('/')
    folder_lst = []

    for f in folders:
        folder_lst.append(f)
        if not os.path.exists('/'.join(folder_lst)):
            os.mkdir('/'.join(folder_lst))

    if type(data) is dict:
        for key in data.keys():
            data2 = data[key]
            if type(data2) is list:
                cols = data2[0].keys()
                file = pd.DataFrame(data2, columns=cols)
                file.to_csv(foldername+'/'+key+'_'+filename+'.csv', mode=
    'w', index=False, header=True, encoding='utf-8-sig')

            else:
                save_tweets(data2, filename, foldername)

    with open(foldername+'/'+filename+'.json', 'w') as f:
        json.dump(data, f)

def main():
    total_tweets = 0

    data = process_yaml()
    bearer_token = create_bearer_token(data)


    start_date = date(2020,1,1)
    end_date = start_date + timedelta(days=1)

    last_date = date.today()

    print('Starting process...')

    while start_date <= last_date:

        # Y-m-dTH:M:SZ
        start_date_str = start_date.strftime('%Y-%m-%dT%H:%M:%SZ')
        end_date_str = end_date.strftime('%Y-%m-%dT%H:%M:%SZ')
```

```
95          start_date_file = start_date.strftime('%Y-%m-%d')
96          start_date_folder = start_date.strftime('%Y-%m-%d')
97
98          url = create_twitter_url(start_date_str, end_date_str)
99          #print('processing URL:', url)
100         res_json = twitter_auth_and_connect(bearer_token, url)
101         if 'data' in res_json.keys():
102             total_tweets += len(res_json['data'])
103             save_tweets(res_json, start_date_file, 'tweets_general/'+
    start_date_folder)
104             print("Total Tweets:", total_tweets, start_date_str)
105             time.sleep(30)
106         else:
107             print("Total Tweets:", total_tweets, start_date_str)
108
109         start_date += timedelta(days=1)
110         end_date = start_date + timedelta(days=1)
111
112     print("Total Tweets:", total_tweets)
113     print('Process has completed.')
114
115 if __name__ == "__main__":
116     main()
```

Appendix B

B.1 Supporting documents related to the application for access to Twitter API

In the following pages it is attached forms, documents and emails related to the application sent to Twitter for access to their product Twitter API, that includes the following documents:

- Application submitted on Twitter for access to Twitter API.
- Confirmation of application received by Twitter.
- Establishing my identity to Twitter.
- Letter from Department sent to establish my identity to Twitter.
- Approval from Twitter to create developer profile and access to API.

Developer Portal

#ResearchProject

When reviewing Academic research applications, it's important to know how you intend on using the Twitter API and Twitter data.

The answers you provide here will help us understand the who, what, why and how of your project. This is critical stuff.

Your answers to these questions illustrate that you have a clearly defined and thought-out Academic research project.

Please answer these questions thoroughly and concisely.

Get help with your application.
Learn more

Basic info > Academic Profile > Project Details > Review > Terms

All fields are required unless marked optional. This info can't be changed once the application has been submitted. If approved, this section will be used to create your Academic Project.

What's your research project's name?

Sentiment Analysis towards the COVID-19 vaccines

Does this project receive funding from outside your academic institution? ⓘ

- ○ Yes
- ◉ No

In English, describe your research project.

The goal of my research project is to identify the general sentiment towards the COVID-19 vaccines in the Republic of Ireland and the UK. This will be analyzed over a specific time (from the announcement of the development of the first vaccine until now*) to understand the evolution of this feeling. I intend to use Neural Network based on Machine Learning** tools for the Sentiment Analysis*** on the texts from tweets. Hence, I am requesting access to the Twitter API to retrieve tweets and subsequent replies/conversations with the words "COVID-19 vaccine", "COVID", "vaccine", and any related words/synonym subjected to this concept.

* By "now" means "by the time I will start the analysis". The duration of my project is 6 months starting from March 2021 to August 2021.
** Refers to the power of computers to learn through data.
*** Refers to the process of determining positive, negative, or neutral feelings.

In English, describe how Twitter data and/or Twitter APIs will be used in your research project.

As the sentiment analysis is a classification problem, the Machine Learning algorithm that will be implemented is based on supervised learning*. Therefore, features (the inputs for my model) and a label (the output) will be identified from the data extracted from Twitter API (after the cleaning process). Once done, the data will be used to train and validate the sentiment analysis model, and then to test how this model will perform on new data. The type of tweets to be used will be current tweets (to test model with new data) and historical tweets (to compare sentiments over different periods).

* Type of learning that maps an input(s) to an output base on example input-output pairs.

Will your research present Twitter data indivually or in aggregate?
Think of it as presenting individual Tweets vs. aggregate statistics or models.

Aggregate

In English, describe your methodology for analyzing Twitter data, Tweets, and/or Twitter users.

To identify the general sentiment towards the COVID-19 vaccines, I intend to use Neural Network based on Machine Learning* tools for the Sentiment Analysis**, and the project would be based on the CRISP-DM* methodology, which would be compound of the following stages:
1. Data Collection. In this stage, I will use Python (possible library to be implemented: tweepy or search-tweets-python) to connect to the API and gather the data. This data would be stored on a local database on my computer with a strong password, to keep the data safe from unauthorized access.
2. Data Preparation. Since social media data is unstructured, it is required to be cleaned before the Modelling stage. This involves a series of tasks like removing irrelevant information such as emojis, special characters, and extra blank spaces. Also, by making format improvements like deleting duplicate tweets or shorten tweets of less than three characters. The next task is to identify features and the label from the cleaned data.
3. Modelling. In this stage, choose a model type (in this case the type is a classification process on Sentiment Analysis to detect sentiment in tweets). The cleaned dataset will be divided to perform the Training, Testing, and Validation processes.
4. Analysis and Deployment. Interpretation of results and design proper visualizations for thesis report (no tweets would be displayed but results from the analysis).

* A structured approach to planning a data mining project.

In English, describe how you will share the outcomes of your research (include tools, data, and/or resources).

1. Research findings may be published through a poster, thesis, and peer-reviewed publications
2. Any processed and collected data related to the project will be disseminated in direct relation to the project and its assessment.
3. The Project deliverables including poster, research thesis, and software product will be stored in the DKIT library after the conclusion of the MSc Research Project.
4. The collected raw data collected will be destroyed at the conclusion of the project i.e. October 2021

Will your analysis make Twitter content or derived information available to a government entity?

- ○ Yes
- ◉ No

Back Next

#ApplicationReceived

Your email **d00242569@student.dkit.ie** has been verified and your application is officially under review!

We'll let you know when it's done, or if we need any additional information from you by sending an email to d00242569@student.dkit.ie.

In the meantime, you can get a head start by learning about the Twitter API by browsing our docs, tutorials, and community forums.

Page 1/1. Confirmation of application received by Twitter.

:

Re: Case# 0198608127: DES Academic Use Case [ref:_00DA0K0A8._5004w274yY3:ref]

Karla Aniela Cepeda Zapata <d00242569@student.dkit.ie>
Mon 3/22/2021 5:55 PM

To: developer-accounts <developer-accounts@twitter.com>
Cc: Rajesh Jaiswal <Rajesh.Jaiswal@dkit.ie>

1 attachments (254 KB)
Karla_ProjectDescriptionV2.pdf;

To whom it may concern,

Please, find the following links to establish my identity. In addition, find attached a letter from the Department of Computing Science and Mathematics.

- https://www.researchgate.net/profile/Karla-Cepeda, this is my ResearchGate with some reports I have written.
- Unfortunately, I had issues when setting up my Google Scholar profile.
- https://www.linkedin.com/in/karla-aniela-cepeda-zapata-28775756/?locale=en_US, my LinkedIn profile.

Also, find a link to my supervisor's Google Scholar account https://scholar.google.com/citations?user=I5CBvwIAAAAJ&hl=en&oi=ao.

Please let me know if you have any questions.

Regards,
Karla Cepeda

From: developer-accounts <developer-accounts@twitter.com>
Sent: Wednesday, March 17, 2021 2:16 PM
To: Karla Aniela Cepeda Zapata <d00242569@student.dkit.ie>
Subject: Case# 0198608127: DES Academic Use Case [ref:_00DA0K0A8._5004w274yY3:ref]

Hello,

Thank you for your request. In order for us to review, we need a few additional details about your plans for the academic access to our API that you're requesting. The information we need is listed below:
Links to webpages that help establish your identity; provide one or more of the following:

- A link to your profile in your institution's faculty or student directory
- A link to your Google Scholar profile

- A link to your research group, lab or departmental website where you are listed

Please reply to this email to provide us this information. Please keep in mind, we need to receive the information listed above within 21 days, or Twitter will close the case without approving access.

Thanks,

Twitter

ref:_00DA0K0A8._5004w274yY3:ref

:

DUNDALK
INSTITUTE OF TECHNOLOGY
INSTITIÚID TEICNEOLAÍOCHTA
DHÚN DEALGAN

Tel: 353 42 9370200
Fax: 353 42 9370201
Web: www.dkit.ie
E-Mail: reception@dkit.ie
E-Mail: first.surname@dkit.ie

19th March 2021

To Whom It May Concern:

Re:	**Karla Aniela Cepeda Zapata**
ID No.:	**D00242569**
Address:	**23 Greenpark Student Accommodation, Dublin Road, Dundalk**
Date of Birth:	**22-OCT-1990**

MASTERS OF SCIENCE IN DATA ANALYTICS

This is to confirm that, **Karla Aniela Cepeda Zapata** is registered as a full time student in the MSc in Data Analytics programme for the academic term 2020/21.

Karla is currently working on a Data Analytics research project for a 30 credits Dissertation module.

Dr. Fiona Lawless,
Head of Department of Computing Science and Mathematics

Scoil na Faisnéisíochta
& na nEalaíon Cruthaitheach
School of Informatics & Creative Arts
Dundalk Institute of Technology
Tel : 042 9370280

Page 1/1. Letter from Department sent to establish my identity to Twitter.

Academic Account Application Approved

Twitter Developer Accounts <developer-accounts@twitter.com>

Mon 3/22/2021 6:14 PM

To: Karla Aniela Cepeda Zapata <d00242569@student.dkit.ie>

Developer

Academic Account Application Approved

Hello,

We're happy to let you know that your request has been approved, and we've enabled your access to utilize the academic level of the Twitter API.

Please complete the setup of your developer profile to get started!

Thanks,

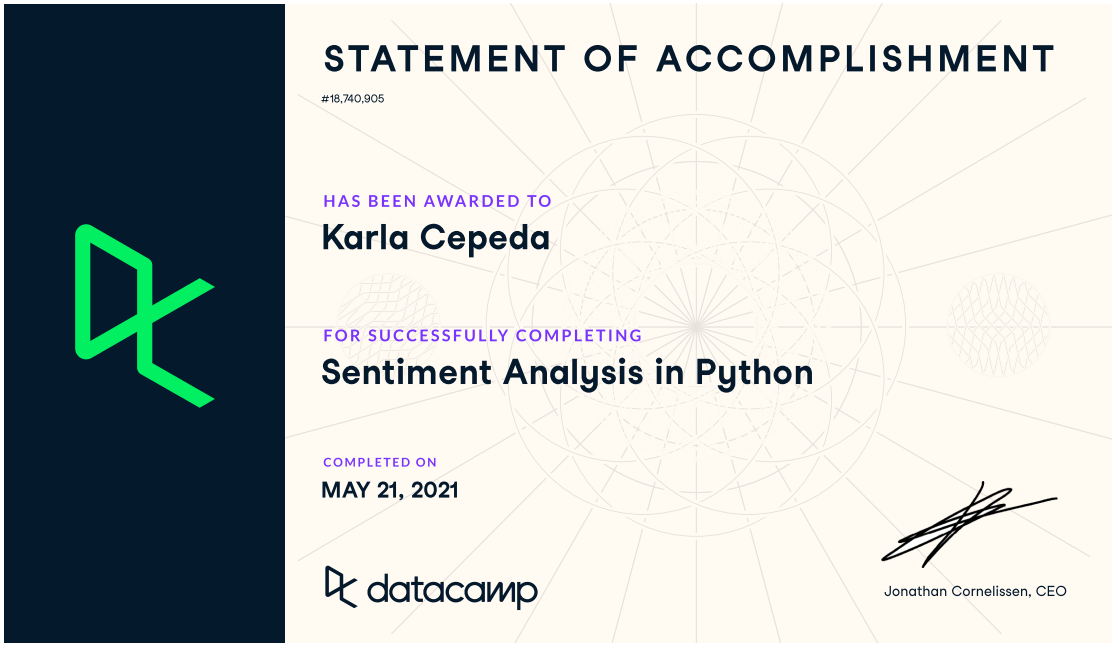
The Twitter Dev team

developer.twitter.com | @twitterdev

Twitter, Inc. 1355 Market Street, San Francisco, CA 94103

Page 1/1. Approval from Twitter to create developer profile.

Appendix C



STATEMENT OF ACCOMPLISHMENT

#18,740,905

HAS BEEN AWARDED TO

Karla Cepeda

FOR SUCCESSFULLY COMPLETING

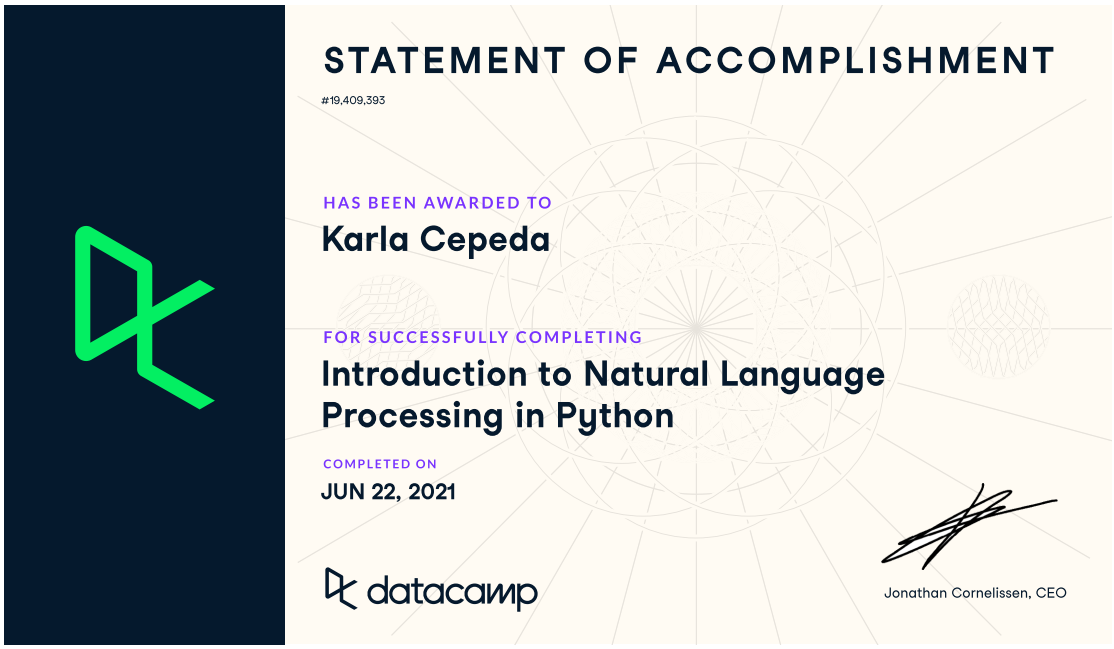
Sentiment Analysis in Python

COMPLETED ON

MAY 21, 2021

datacamp

Jonathan Cornelissen, CEO



STATEMENT OF ACCOMPLISHMENT

#19,409,393

HAS BEEN AWARDED TO

Karla Cepeda

FOR SUCCESSFULLY COMPLETING

Introduction to Natural Language Processing in Python

COMPLETED ON

JUN 22, 2021

datacamp

Jonathan Cornelissen, CEO

STATEMENT OF ACCOMPLISHMENT

#19,839,826

HAS BEEN AWARDED TO

Karla Cepeda

FOR SUCCESSFULLY COMPLETING

Advanced NLP with spaCy

COMPLETED ON

JUN 23, 2021

datacamp

Jonathan Cornelissen, CEO

STATEMENT OF ACCOMPLISHMENT

#19,851,739

HAS BEEN AWARDED TO

Karla Cepeda

FOR SUCCESSFULLY COMPLETING

Feature Engineering for NLP in Python

COMPLETED ON

JUN 24, 2021

datacamp

Jonathan Cornelissen, CEO

Appendix D

D.1 Proposed Analysis

In the following, it is enlisted the plan that taken for this project:

- 21/06/2021 to 25/06/2021. **Natural Processing Language Courses**
  Introduction to Natural Processing Language in Python.
  Advanced NPL with spaCy.
  Feature Engineering for NPL in Python.
  Total time: roughly 4hrs per course.
- 28/06/2021 to 01/07/2021. **Covid-19 critical dates**
  Look into key dates related to Covid-19 in Ireland. May include international events regarding Covid-19.
- 05/07/2021 to 08/07/2021. **Recollection of Data**
  Add to the keyword list alternative and official names for vaccines such as Vaxzevria which was not included in the previous.
  As code is already done, this task would be easier, however, it takes time.
- 09/07/2021 to 11/07/2021. **Analysis on one-word and two-word tweets length**
  This should be done by analyzing the most frequent words by creating a cloud word or a graph to stand out the most used words. This will be used to decide which one would stay and what to remove.
- 12/07/2021 to 20/07/2021. **Research on labeling process**
  Look into how to label tweets.
  Look into sentiment analyzer tools available online (free and pricing options) for the labeling process. Options: Azure (as this is used in a tutorial from Twitter API) and Monkey Learn.
- 19/07/2021 to 26/07/2021. **labeling process using online tools**
  Labeling tweets with online tools and model made.
- 23/07/2021 to 27/07/2021. **Modeling process**
  Build and Train a model for the Sentiment Analysis classification process, using Machine Learning tools.
- 30/07/2021 to 06/08/2021. **Analysis and Research Report**
  Answer Research questions enlisted in table 1.1.
- 07/07/2021 to 14/08/2021. **Daily Tweet Collector**
  Design Python script to retrieve daily tweets related to
- 16/07/2021 to 23/08/2021. **Writing final report**
  Describe everything that has been done.

- 3/09/2021. **Submition for feedback**
  Submit the final draft of your thesis to your supervisor for feedback.
- 10/09/2021. **Supporting documents Submition**
  Submit your supporting documents or artifacts (as indicated in section 2 in the Final Dissertation Guidance document) on Moodle as a ZIP file following the naming convention yourSurname_final_artifacts.zip.
- 14/09/2021. **Final Submition**
  Submit your Final Thesis on Moodle in PDF only following the naming convention yourSurname_final_report.pdf.

Bibliography

Al-Shabi, M. (2020), 'Evaluating the performance of the most important lexicons used to sentiment analysis and opinions mining'.

Angiani, G., Ferrari, L., Fontanini, T., Fornacciari, P., Iotti, E., Magliani, F. & Manicardi, S. (2016), 'A comparison between preprocessing techniques for sentiment analysis in twitter'.

Ashis, P. (2012), 'Support vector machine-a survey', *IJETAE* **2**.

Balakrishnan, V. & Ethel, L.-Y. (2014), 'Stemming and lemmatization: A comparison of retrieval performances'.

BBC News (2020), 'Coronavirus: First case confirmed in republic of ireland'. Available from: *https://www.bbc.com/news/world-europe-51693259* [accessed: 04.06.2021].

Berger, A. L., Della Pietra, S. A. & Della Pietra, V. J. (2002), 'A maximum entropy approach to natural language processing', *Computational Linguistics* **22**.

Bhattacharyya, S. (2018), 'Ridge and lasso regression: L1 and l2 regularization'. Available from: *https://towardsdatascience.com/ridge-and-lasso-regression-a-complete-guide-with-python-scikit-learn-e20e34bcbf0b* [accessed: 25.08.2021].

Bhuta, S., Doshi, A., Doshi, U. & M., N. (2014), 'A review of techniques for sentiment analysis of twitter data', *International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)* pp. 583–591.

BioSpace (2020), 'A timeline of covid-19 vaccine development.'. Available from: *https://www.biospace.com/article/a-timeline-of-covid-19-vaccine-development* [accessed: 02.06.2021].

Bonta, V., Kumaresh, N. & Janardhan, N. (2019), 'A comprehensive study on lexicon based approaches for sentiment analysis', *Asian Journal of Computer Science and Technology* **8**, 1–6.

Bronchal, L. (2017), 'Sentiment analysis with svm'. Available from: *https://www.kaggle.com/lbronchal/sentiment-analysis-with-svm/data* [accessed: 25.08.2021].

Brownlee, J. (2017), 'A gentle introduction to the bag-of-words model'. Available from: *https://machinelearningmastery.com/gentle-introduction-bag-words-model/* [accessed: 11.06.2021].

Brownlee, J. (2020), 'Repeated k-fold cross-validation for model evaluation in python'. Available from: *https://machinelearningmastery.com/repeated-k-fold-cross-validation-with-python/* [accessed: 25.08.2021].

Burke-Kennedy, E. (2021), 'Tough lockdown measures keep unemployment rate at 24'. Available from: *https://www.irishtimes.com/business/economy/tough-lockdown-measures-keep-unemployment-rate-at-24-1.4524967* [accessed: 04.06.2021].

Centers for Disease Control and Prevention - USA (2012), 'Vaccines: The basics'. Available from: *https://www.cdc.gov/vaccines/vpd/vpd-vac-basics.html* [accessed: 02.06.2021].

Centers for Disease Control and Prevention - USA (2018), 'Immunization: The basics'. Available from: *https://www.cdc.gov/vaccines/vac-gen/imz-basics.htm* [accessed: 02.06.2021].

Chance, D., McQuinn, C. & Walsh, A. (n.d.), 'Coronavirus ireland: Country set for deep recession as economy to shrink by 10pc and mass unemployment grows'.

Cullen, P. (2020), 'Coronavirus cases now confirmed in every county in ireland'. Available from: *https://www.irishtimes.com/news/health/coronavirus-cases-now-confirmed-in-every-county-in-ireland-1.4209389* [accessed: 04.06.2021].

Cullen, P. & Hilliard, M. (2021), 'More than 50,000 children aged 12-15 registered to receive covid-19 vaccine'. Available from: https://www.irishtimes.com/news/health/more-than-50-000-children-aged-12-15-registered-to-receive-covid-19-vaccine-1.4645645/ [accessed: 20.08.2021].

Curran, I. (2020), 'Covid-19 vaccine rollout begins today at four irish hospitals'. Available from: *https://www.thejournal.ie/vaccine-rollout-begins-5312188-Dec2020/* [accessed: 04.06.2021].

Dai, X., Xiong, Y., Li, N. & Jian, C. (2001), 'Vaccine types', *IntechOpen* pp. 1–9.

DataCamp, Inc. (2019), 'Sentiment analysis in python'. Available from: *https://www.datacamp.com/courses/sentiment-analysis-in-python* [accessed: 11.06.2021].

DataCamp, Inc. (2021*a*), 'About'. Available from: *https://www.datacamp.com/about* [accessed: 11.06.2021].

DataCamp, Inc. (2021*b*), 'Career-building learning paths'. Available from: *https://www.datacamp.com/tracks/career* [accessed: 11.06.2021].

DataCamp, Inc. (2021*c*), 'Data skill learning paths'. Available from: *https://www.datacamp.com/tracks/skill* [accessed: 11.06.2021].

Dekanovsky, V. (2021), 'Complete guide to python's cross-validation with examples'. Available from: *https://towardsdatascience.com/complete-guide-to-pythons-cross-validation-with-examples-a9676b5cac12* [accessed: 25.08.2021].

Dey, A. (2016), 'Machine learning algorithms: A review', *International Journal of Computer Science and Information Technologies (IJCSIT)* **7**(3), 1174–1179.

Digital Desk Staff (2021), 'Who warns of fourth wave in ireland with increase in large gatherings'. Available from: *https://www.breakingnews.ie/ireland/who-warns-of-fourth-wave-in-ireland-with-increase-in-large-gatherings-1136663.html* [accessed: 04.06.2021].

Du, J., Vong, C.-M. & Chen, C. L. P. (2021), 'Novel efficient rnn and lstm-like architectures: Recurrent and gated broad learning systems and their applications for text classification', *IEEE Transactions on Cybernetics* **51**(3), 1586–1597.

Dwyer, O. (2020), '79-year-old dublin woman first in republic of ireland to get covid-19 vaccine'. Available from: *https://www.thejournal.ie/first-coronavirus-vaccine-ireland-5312217-Dec2020/* [accessed 31.08.2021].

Dwyer, O. (2021), 'At a glance: Everything you need to know about the re-opening plan announced tonight'. Available from: *https://www.thejournal.ie/main-points-lockdown-easing-ireland-may-5424025-Apr2021/* [accessed: 04.06.2021].

Elouardighi, A., Maghfour, M., Hammia, H. & Aazi, F.-z. (2017), 'A machine learning approach for sentiment analysis in the standard or dialectal arabic facebook comments', pp. 1–8.

Eremyan, R. (2020), 'Four pitfalls of sentiment analysis accuracy'. Available from: *https://www.toptal.com/deep-learning/4-sentiment-analysis-accuracy-traps* [accessed: 11.06.2021].

European Centre for Disease Prevention and Control (2021), 'Timeline of ecdc's response to covid-19'. Available from: *https://www.ecdc.europa.eu/en/covid-19/timeline-ecdc-response* [accessed: 25.05.2021].

European Commission (2021), 'Safe covid-19 vaccines for europeans'. Available from: *https://ec.europa.eu/info/live-work-travel-eu/coronavirus-response/safe-covid-19-vaccines-europeans_en* [accessed: 03.06.2021].

European Medicines Agency (2021), 'Conditional marketing authorisation'. Available from: *https://www.ema.europa.eu/en/human-regulatory/marketing-authorisation/conditional-marketing-authorisation* [accessed: 03.06.2021].

Fauci, A. S., Lane, H. C. & Redfield, R. R. (2020), 'Covid-19 - navigating the uncharted', *The New England Journal of Medicine* **382**(13), 1268–1269.

Forbes (2019), 'Data is the new oil – and that's a good thing'. Available from: *https://www.forbes.com/sites/forbestechcouncil/2019/11/15/data-is-the-new-oil-and-thats-a-good-thing/* [accessed: 02.06.2021].

Gandhi, R. (2018), 'Support vector machine — introduction to machine learning algorithms'. Available from: *https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47* [accessed: 02.06.2021].

Global Citizen - Gavi The Vaccine Alliance (2020), 'Ask an expert: Why are there so many covid-19 vaccines - and is it better to have more?'. Available from: *https://www.gavi.org/vaccineswork/ask-expert-why-are-there-so-many-covid-19-vaccines-and-it-better-have-more* [accessed: 11.06.2021].

Go, A., Bhayani, R. & Huang, L. (2009), 'Twitter sentiment classification using distant supervision', *Processing* **150**.

gov.ie (2020), 'Access covid-19 data on ireland's open data portal'. Available from: *https://www.gov.ie/en/service/0da52-access-covid-19-coronavirus-data-on-irelands-open-data-portal/* [accessed: 04.06.2021].

gov.ie (2021), 'Your guide to the changes - resilience and recovery: The path ahead'. Available from: *https://www.gov.ie/en/press-release/7894b-post-cabinet-statement-resilience-and-recovery-the-path-ahead/?referrer=http://www.gov.ie/en/press-release/0bd80-new-public-health-measures-announced-the-path-ahead/* [accessed: 04.06.2021].

Hassan, S., Sheikh, F., Jamal, S., Ezeh, J. & Akhtar, A. (2020), 'Coronavirus (covid-19): A review of clinical feature, diagnosis, and treatment', *Cureus* **12**(3), 1268–1269.

Haynes, B., Corey, L., Fernandes, P., Gilbert, P., Hotez, P., Rao, S., Santos, M., Schuitemaker, H., Watson, M. & Arvin, A. (2020), 'Prospects for a safe covid-19 vaccine.', *Cureus* **12**(568).

Health Service Executive (2021), 'Symptoms of covid-19'. Available from: *https://www2.hse.ie/conditions/covid19/symptoms/overview/* [accessed: 1.06.2021].

Hearst, M., Dumais, S., Osman, E., Platt, J. & Scholkopf, B. (1998), 'Support vector machines', *Intelligent Systems and their Applications, IEEE* **13**, 18–28.

Health Protection Surveillance Centre (2020), 'About HPSC'. Available from: *https://www.hpsc.ie/abouthpsc/* [accessed: 04.06.2021].

Health Service Executive (2020), 'Covid tracker app now available to download'. Available from: *https://healthservice.hse.ie/staff/news/coronavirus/covid-tracker-app-now-available-to-download.html* [accessed: 04.06.2021].

Health Service Executive (2021), 'Rollout of Covid-19 vaccines in ireland'. Available from: *https://www2.hse.ie/screening-and-vaccinations/covid-19-vaccine/rollout/* [accessed: 04.06.2021].

HSE Press (2020), 'Annie lynch is the first person to receive the pfizer biontech covid19 vaccine in ireland'. Available from: *https://www.hse.ie/eng/services/news/media/pressrel/annie-lynch-is-the-first-person-to-receive-the-pfizer-biontech-covid19-vaccine-in-ireland.html* [accessed: 04.06.2021].

Health Service Executive, Government of Ireland (2020), 'Covid tracker app'. Available from: *https://covidtracker.ie/* [accessed: 04.06.2021].

IBM (2020*a*), 'What is machine learning?'. Available from: *https://www.ibm.com/cloud/learn/machine-learning* [accessed: 02.06.2021].

IBM (2020*b*), 'What is natural language processing?'. Available from: *https://www.ibm.com/cloud/learn/natural-language-processing* [accessed: 02.06.2021].

IBM (2021), 'CRISP-DM - help overview'. Available from: *https://www.ibm.com/docs/en/spss-modeler/SaaS?topic=dm-crisp-help-overview* [accessed: 29.05.2021].

IBM Cloud Team, IBM Cloud (2021), 'Python vs. r: What's the difference?'. Availabe from: *https://www.ibm.com/cloud/blog/python-vs-r* [accessed: 27.07.2021].

Joachims, T. (1998), 'Text categorization with support vector machines', *Proc. European Conf. Machine Learning (ECML'98)* .

Joshi, P. (2018), 'Comprehensive hands on guide to twitter sentiment analysis with dataset and code'. Available from: *https://www.analyticsvidhya.com/blog/2018/07/hands-on-sentiment-analysis-dataset-python/* [accessed: 07.06.2021].

Kaggle (2020), '2020 kaggle machine learning data science survey'. Available from: *https://www.kaggle.com/c/kaggle-survey-2020* [accessed: 27.07.2021].

Kashte, S., Gulbake, A., El-Amin Iii, S. & Gupta, A. (2021), 'Covid-19 vaccines: rapid development, implications, challenges and future prospects.', *Hum Cell* **34**(3), 711–733.

Kaur, S. P. & Gupta, V. (2020), 'Covid-19 vaccine: A comprehensive status report'.

Khafaji, H. & Habeeb, A. (2017), 'Efficient algorithms for preprocessing and stemming of tweets in a sentiment analysis system'.

Khuroo, M. S., Khuroo, M., Khuroo, M. S., Sofi, A. A. & Khuroo, N. S. (2020), 'Covid-19 vaccines: A race against time in the middle of death and devastation!', *Elsevier* **10**(6), 610–621.

Li, H., Liu, S., Yu, X., Tang, S. & Tang, C. (2020), 'Coronavirus disease 2019 (covid-19): current status and future perspectives.', *Int J Antimicrob Agents* **55**(5).

Maguire, R. (2021), 'What is monkeylearn?'. Available from: *https://help.monkeylearn.com/en/articles/2174206-what-is-monkeylearn* [accessed: 18.08.2021].

Maharani, W. (2013), 'Microblogging sentiment analysis with lexical based and machine learning approaches', pp. 439–443.

Malik, U. (2019), 'Python for nlp: Movie Sentiment Analysis using Deep Learning in Keras'. Available from: *https://stackabuse.com/python-for-nlp-movie-sentiment-analysis-using-deep-learning-in-keras* [accessed: 04.06.2021].

Mandloi, L. & Patel, R. (2020), 'Twitter sentiments analysis using machine learninig methods', pp. 1–5.

Merriam-Webster (2021*a*), 'Definition of microblogging'. Available from: *https://www.merriam-webster.com/dictionary/microblogging* [accessed: 26.05.2021].

Merriam-Webster (2021*b*), 'Defintion of irony'. Available from: *https://www.merriam-webster.com/dictionary/irony* [accessed: 11.06.2021].

Metwalli, S. A. (2020), 'Nlp 101: What is natural language processing?'. Available from: *https://towardsdatascience.com/nlp-101-what-is-natural-language-processing-b4a968a3b7bf* [accessed: 02.06.2021].

Microsoft Corporation (2021*a*), 'Create your azure free account today'. Available from: *https://azure.microsoft.com/en-us/free/search/* [accessed: 18.08.2021].

Microsoft Corporation (2021*b*), 'Data, privacy, and security for text analytics'. Available from: *https://docs.microsoft.com/en-us/legal/cognitive-services/text-analytics/data-privacy?context=/azure/cognitive-services/text-analytics/context/context* [accessed: 18.08.2021].

Microsoft Corporation (2021*c*), 'Text analytics'. Available from: *https://azure.microsoft.com/en-us/services/cognitive-services/text-analytics/#overview* [accessed: 18.08.2021].

Microsoft Corporation (2021*d*), 'Transparency note for sentiment analysis'. Available from: *https://docs.microsoft.com/en-us/legal/cognitive-services/text-analytics/transparency-note-sentiment-analysis?context=/azure/cognitive-services/text-analytics/context/context* [accessed: 18.08.2021].

Microsoft Corporation (2021*e*), 'What is azure?'. Available from: *https://azure.microsoft.com/en-us/overview/what-is-azure/* [accessed: 18.08.2021].

Microsoft Corporation (2021*f*), 'What is the text analytics api?'. Available from: *https://docs.microsoft.com/en-us/azure/cognitive-services/text-analytics/overview#data-limits* [accessed: 18.08.2021].

Mohammad, A.-S., Omar, Q., Mahmoud, A.-A., Yaser, J. & Brij, G. (2018), 'Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of arabic hotels' reviews', *Journal of Computational Science* **27**, 386–393.
URL: *https://www.sciencedirect.com/science/article/pii/S1877750317305252*

Mohri, M., Rostamizadeh, A. & Talwalkar, A. (2018), 'Foundations of machine learning'.

Monkey Learn (n.d.), 'Sentiment analysis: A definitive guide.'. Available from: *https://monkeylearn.com/sentiment-analysis/* [accessed: 02.06.2021].

MonkeyLearn Inc. (2019), 'Privacy policy'. Available from: *https://monkeylearn.com/privacy* [accessed: 18.08.2021].

Muller, A. & Guido, S. (2017), 'Introduction to machine learning with python'.

Neethu, M. S. & Rajasree, R. (2013), 'Sentiment analysis in twitter using machine learning techniques', pp. 1–5.

Nigam, K., Lafferty, J. & McCallum, A. (1999), 'Using maximum entropy for text classification', *School of Computer Science* pp. 61–67.

O'Regan, E. & Collins, S. (2021), 'Ireland to receive extra 545,000 pfizer vaccine doses between april and june'. Available from: https://www.independent.ie/world-news/coronavirus/ireland-to-receive-extra-545000-pfizer-vaccine-doses-between-april-and-june-40312599.html [accessed: 27.08.2021].

Pang, B., Lee, L. & Vaithyanathan, S. (2002), 'Thumbs up? sentiment classification using machine learning techniques', pp. 79–86.
URL: *https://aclanthology.org/W02-1011*

Paruchuri, V. (2015), 'Tutorial: Predicting movie review sentiment with naive bayes'. Available from: https://www.dataquest.io/blog/naive-bayes-tutorial/ [accessed: 17.08.2021].

Paul, R. (2020), 'Tf-idf with scikit learn'. Available from: *https://medium.com/analytics-vidhya/tf-idf-with-scikit-learn-e73963eda5e3* [accessed: 23.08.2021].

Perumal, V., Curran, T. & Hunter, M. (2020), 'First case of covid-19 in ireland', *Ulster Medical Society* **89**(2), 128.

Pollard, D. (1997), 'Probability theory - conditional probability'. Department of Statistics, Yale University. Available from: *http://www.stat.yale.edu/Courses/1997-98/101/condprob.htm* [accessed: 03.06.2021].

Pykes, K. (2021), 'Fighting overfitting with l1 or l2 regularization - which one is better?'. Available from: *https://neptune.ai/blog/fighting-overfitting-with-l1-or-l2-regularization* [accessed: 25.08.2021].

Raghunathan, D. (2020), 'Nlp in python- vectorizing'. Available from: *https://towardsdatascience.com/nlp-in-python-vectorizing-a2b4fc1a339e* [accessed: 23.08.2021].

Ray, S. (2017). Available from: *https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/* [accesed: 18.08.2021].

RTE (2020*a*), 'First batch of pfizer-biontech vaccine arrives in ireland'. Available from: *https://www.rte.ie/news/2020/1226/1186461-coronavirus-ireland/* [accessed: 20.08.2021].

RTE (2020*b*), 'New restrictions: Exceptions for leaving your home'. Available from: *https://www.rte.ie/news/2020/0327/1126911-ireland-restrictions-covid19/* [accessed: 04.06.2021].

RTE (2020*c*), 'Vaccine decision based on science, education minister tells teachers'. Available from: *https://www.rte.ie/news/education/2021/0405/1208067-teachers-conferences/* [accessed: 20.08.2021].

Rudge, P., Ratcliff, G., Lentz, T. L., Haines, D. E., Matthews, P. B., Nathan, P. W., Loewy, A. D. & Noback, C. R. (2020), 'Human nervous system - Receptors', *Encyclopedia Britannica* . Available from: https://www.britannica.com/science/human-nervous-system [accessed: 04.06.2021].

Rustam, F., Khalid, M., Aslam, W., Rupapara, V., Mehmood, A. & Choi, G. S. (2021), 'A performance comparison of supervised machine learning models for covid-19 tweets sentiment analysis', *PloS one* .
URL: *https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7906356/*

Shah, P. (2020), 'Basic tweet preprocessing in python'. Available from: *https://towardsdatascience.com/basic-tweet-preprocessing-in-python-efd8360d529e* [accessed: 07.06.2021].

Siddiqui, S. (2019), 'How would we find a better activation function than relu?'. Available from: *https://medium.com/shallow-thoughts-about-deep-learning/how-would-we-find-a-better-activation-function-than-relu-4409df217a5c* [accessed: 04.06.2021].

sklearn (2021*a*), '3.1. cross-validation: Evaluating estimator performance'. Available from: *https://scikit-learn.org/stable/modules/cross_validation.html#repeated-k-fold* [accessed: 25.08.2021].

sklearn (2021*b*), '3.3. metrics and scoring: Quantifying the quality of predictions'. Available from: *https://scikit-learn.org/stable/modules/model_evaluation.html#accuracy-score* [accessed: 24.08.2021].

sklearn (2021*c*), '3.4. validation curves: Plotting scores to evaluate models'. Available from: *https://scikit-learn.org/stable/modules/learning_curve.html#validation-curve*, journal=scikit [accessed: 24.08.2021].

sklearn (2021*d*), '6.1. pipelines and composite estimators'. Available from: *https://scikit-learn.org/stable/modules/compose.html#combining-estimators* [accessed: 23.08.2021].

sklearn (2021*e*), '8.26.1.2. sklearn.svm.linearsvc'. Available from: *https://ogrisel.github.io/scikit-learn.org/sklearn-tutorial/modules/generated/sklearn.svm.LinearSVC.html* [accessed: 23.08.2021].

sklearn (2021*f*), 'Confusion matrix'. Available from: *https://scikit-learn.org/stable/auto_examples/model_selection/plot_confusion_matrix.html#sphx-glr-auto-examples-model-selection-plot-confusion-matrix-py* [accessed: 24.08.2021].

sklearn (2021*g*), 'sklearn.linear_model.sgdclassifier'. Available from: *https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html?highlight=sgdclassifier#sklearn.linear_model.SGDClassifier* [accessed: 25.08.2021].

sklearn (2021*h*), 'sklearn.model_selection.stratifiedkfold'. Available from: *https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.StratifiedKFold.html?highlight=stratifiedkfold#sklearn.model_selection.StratifiedKFold* [accessed: 25.08.2021].

sklearn (2021*i*), 'sklearn.svm.svr'. Available from: *https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html#sklearn.svm.SVR* [accessed: 23.08.2021].

sklearn (2021*j*), 'Working with text data'. Available from: *https://scikit-learn.org/stable/tutorial/text_analytics/working_with_text_data.html* [accessed: 24.08.2021].

Sky News (2020), 'Covid-19 vaccine: Historic moment first uk patient receives coronavirus pfizer/biontech coronavirus jab outside clinical trial'. Available from: *https://news.sky.com/story/covid-19-uk-patient-becomes-first-in-world-to-receive-pfizer-biontech-coronavirus-vaccine-outside-a-clinica* [accessed: 04.06.2021].

Stokx, J. (2019), 'Defining unmet medical need'. Available from: *https://www.ema.europa.eu/en/documents/presentation/presentation-defining-unmet-medical-need-jstokx_en.pdf* [accessed: 03.06.2021].

Taboada, M., Brooke, J., Tofiloski, M., Voll, K. & Stede, M. (2011), 'Lexiconbased methods for sentiment analysis', *MIT Press* **32**(2), 267–307.

Talpada, H., Halgamuge, M. & Tran Quoc Vinh, N. (2019), 'An analysis on use of deep learning and lexical-semantic based sentiment analysis method on twitter data to understand the demographic trend of telemedicine', pp. 1–9.

timeanddate.com (n.d.), 'Is chinese new year a public holiday?'. Available from: *https://www.timeanddate.com/holidays/china/spring-festival* [accessed: 26.05.2021].

Tss, R. & Andrade, C. (2011), 'The mmr vaccine and autism: Sensation, refutation, retraction, and fraud'.

Twitter Inc. (2017). Available from: *https://github.com/twitter-forks/mysql* [accessed 2021.08.22].

Twitter, Inc. (2020*a*), 'Developer terms - developer agreement'. Available from: *https://developer.twitter.com/en/developer-terms/agreement* [accessed: 06.06.2021].

Twitter, Inc. (2020*b*), 'Get started with the twitter developer platform'. Available from: *https://developer.twitter.com/en/docs/getting-started* [accessed: 26.05.2021].

Twitter, Inc. (2020*c*), 'Oauth 2.0 bearer token'. Available from: *https://developer.twitter.com/en/docs/authentication/oauth-2-0* [accessed: 27.05.2021].

Twitter, Inc. (2020*d*), 'Twitter terms of service'. Available from: *https://twitter.com/en/tos* [accessed: 06.06.2021].

Twitter, Inc. (2021*a*), 'About the twitter api'. Available from: *https://developer.twitter.com/en/docs/twitter-api/getting-started/about-twitter-api* [accessed: 27.05.2021].

Twitter, Inc. (2021*b*), 'Account activity api'. Available from: *https://developer.twitter.com/en/docs/twitter-api/enterprise/account-activity-api/guides/getting-started-with-webhooks* [accessed: 29.05.2021].

Twitter Inc. (2021*c*), 'Building queries for search tweets'. Available from: *https://developer.twitter.com/en/docs/twitter-api/tweets/search/integrate/build-a-query* [accessed: 01.06.2021].

Twitter, Inc. (2021*d*), 'Getting access to the twitter api'. Available from: *https://developer.twitter.com/en/docs/twitter-api/getting-started/getting-access-to-the-twitter-api* [accessed: 27.05.2021].

Twitter, Inc. (2021*e*), 'Rate limits: Standard v1.1'. Available from: *https://developer.twitter.com/en/docs/twitter-api/v1/rate-limits* [accessed: 29.05.2021].

Twitter, Inc. (2021*f*), 'Search tweets'. Available from: *https://developer.twitter.com/en/docs/twitter-api/v1/tweets/search/overview* [accessed: 29.05.2021].

Twitter, Inc. (2021*g*), 'Search tweets'. Available from: *https://developer.twitter.com/en/docs/twitter-api/tweets/search/quick-start/recent-search* [accessed: 29.05.2021].

Twitter, Inc. (2021*h*), 'Tweet lookup'. Available from: *https://developer.twitter.com/en/docs/twitter-api/tweets/lookup/quick-start* [accessed: 29.05.2021].

Twitter, Inc. (2021*i*), 'Twitter api'. Available from: *https://developer.twitter.com/en/docs/twitter-api* [accessed: 26.05.2021].

Twitter, Inc. (2021*j*), 'Twitter api v2: Early access'. Available from: *https://developer.twitter.com/en/docs/twitter-api/early-access* [accessed: 29.05.2021].

Twitter, Inc. (2021*k*), 'Versioning'. Available from: *https://developer.twitter.com/en/docs/twitter-api/versioning* [accessed: 29.05.2021].

UK Reserach and Innovation (2020), 'What is coronavirus? the different types of coronaviruses'. Available from: *https://coronavirusexplained.ukri.org/en/article/cad0003/* [accessed: 26.05.2021].

U.S. Food and Drug Administration (2018), 'What we do'. Available from: *https://www.fda.gov/about-fda/what-we-do* [accessed: 02.06.2021].

Wagner, A. L. (2020), 'What makes a 'wave' of disease? an epidemiologist explains'. Available from: *https://theconversation.com/what-makes-a-wave-of-disease-an-epidemiologist-explains-141573* [accessed: 04.06.2021].

wellcome (2021), 'How have covid-19 vaccines been made quickly and safely?'. Available from: *https://wellcome.org/news/quick-safe-covid-vaccine-development* [accessed: 11.06.2021].

World Health Organization (2021), 'Covid-19 vaccine tracker and landscape'. Available from: *https://www.who.int/publications/m/item/draft-landscape-of-covid-19-candidate-vaccines* [accessed: 11.06.2021].

WHO/N.K. Acquah (2021), 'Covax'. Available from: *https://www.who.int/initiatives/act-accelerator/covax* [accessed: 11.06.2021].

Wikipedia The Free Encyclopedia (2021), 'Twitter'. Available from: *https://en.wikipedia.org/wiki/Twitter* [accessed: 26.05.2021].

Williams, R., Jindal, N. & Batra, A. (2019), 'Sentiment analysis using lexicon based approach', *IITM Journal of Management and IT* pp. 68–76.

Wolff, R. (2020), 'Sentiment analysis with machine learning: Process tutorial'. Available from: *https://monkeylearn.com/blog/sentiment-analysis-machine-learning/* [accessed: 07.06.2021].

Woodruff, A. (2019), 'What is a neuron?'. *University of Queensland*. Available from: *https://qbi.uq.edu.au/brain/brain-anatomy/what-neuron* [accessed: 04.06.2021].

World Health Organization (2020*a*), 'Naming the coronavirus disease (covid-19) and the virus that causes it'. Available from: *https://www.who.int/emergencies/diseases/novel-coronavirus-2019/technical-guidance/naming-the-coronavirus-disease-(covid-2019)-and-the-virus-that-causes-it* [accessed: 25.05.2021].

World Health Organization (2020*b*), 'Novel coronavirus(2019-ncov) situation report – 22'. Available from: *https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200211-sitrep-22-ncov.pdf?sfvrsn=fb6d49b1_2* [accessed: 25.05.2021].

World Health Organization (2020*c*), 'Who director-generalś opening remarks at the media briefing on covid-19 - 11 march 2020'. Available from: *https://www.who.int/director-general/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---11-march-2020* [accessed: 25.05.2021].

World Health Organization (2021*a*), 'Coronavirus disease (covid-19) advice for the public'. Available from: *https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public* [accessed: 26.05.2021].

World Health Organization (2021*b*), 'Coronavirus disease (covid-19): How is it transmitted?'. Available from: *https://www.who.int/news-room/q-a-detail/coronavirus-disease-covid-19-how-is-it-transmitted* [accessed: 26.05.2021].

World Health Organization (2021*c*), 'WHO coronavirus (covid-19) dashboard'. Available from: *https://covid19.who.int/* [accessed: 26.05.2021].

Zhang, L., Ghosh, R., Dekhil, M., Hsu, M. & Liu, B. (2011), 'Combining lexicon-based and learning-based methods for twitter sentiment analysis', *HP Laboratories* .

Zhang, Y., Xu, J., Li, H. & Cao, B. (2020), 'A novel coronavirus (covid-19) outbreak a call for action', *Chest* **157**(4), 99–100.