

# Deep Word Embedding based Sentimental Analysis of COVID-19 Tweets



Author

Sobia Qaiser

00000320760

Supervisor

Dr. Muhammad Usman Akram

DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING  
COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING  
NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

June 2023

Deep Word Embedding based Sentimental Analysis of COVID-19  
Tweets

Author

Sobia Qaiser

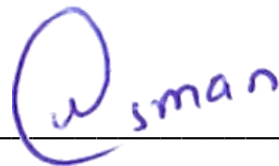
MS19CSE(SE)00000320760

A thesis submitted in partial fulfillment of the requirements for the degree of  
MS Software Engineering

Thesis Supervisor

Dr. Muhammad Usman Akram

Thesis Supervisor's Signature: \_\_\_\_\_



DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING  
COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING  
NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY,  
ISLAMABAD

June, 2023

**THESIS ACCEPTANCE CERTIFICATE**

Certified that final copy of MS/MPhil thesis written by **NS Sobia Qaiser** Registration No. 00000320760, of College of E&ME has been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial fulfillment for award of MS/MPhil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the thesis.

Signature : 

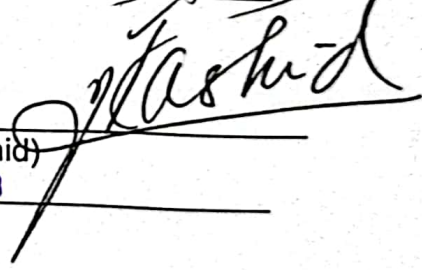
Name of Supervisor: **Dr Muhammad Usman Akram**

Date: 06-06-2023

Signature of HOD:  
(Dr Usman Qamar)

Date: 06-06-2023 

Signature of Dean:  
(Brig Dr Nasir Rashid)

Date: 10 6 JUN 2023 

## Declaration

I certify that this research work titled “*Deep Word Embedding based Sentimental Analysis of COVID-19 Tweets*” is my own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources has been properly acknowledged / referred.



Signature of Student

Sobia Qaiser

MS19CSE(SE)00000320760

## Language Correctness Certificate

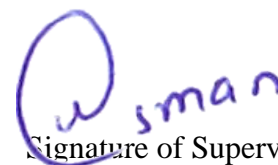
This thesis has been read by an English expert and is free of typing, syntax, semantic, grammatical, and spelling mistakes. Thesis is also according to the format given by the university.



Signature of Student

Sobia Qaiser

MS19CSE(SE)00000320760



Signature of Supervisor

Dr. Muhammad Usman Akram

## **Copyright Statement**

- Copyright in text of this thesis rests with the student author. Copies (by any process) either in full, or of extracts, may be made only in accordance with instructions given by the author and lodged in the Library of NUST College of E&ME. Details may be obtained by the Librarian. This page must form part of any such copies made. Further copies (by any process) may not be made without the permission (in writing) of the author.
- The ownership of any intellectual property rights which may be described in this thesis is vested in NUST College of E&ME, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the College of E&ME, which will prescribe the terms and conditions of any such agreement.
- Further information on the conditions under which disclosures and exploitation may take place is available from the Library of NUST College of E&ME, Rawalpindi.

## **Acknowledgements**

All praise and glory to Almighty Allah (the most glorified, the most high) who gave me the courage, patience, knowledge and ability to carry out this work and to persevere and complete it satisfactorily. Undoubtedly, HE eased my way and without HIS blessings I can achieve nothing.

I would like to express my sincere gratitude to my advisor Dr. Muhammad Usman Akram for boosting my morale and for his continual assistance, motivation, dedication and invaluable guidance in my quest for knowledge. I am blessed to have such a co-operative advisor and kind mentor for my research.

Along with my advisor, I would like to acknowledge my entire thesis committee: Dr. Arslan Shaukat and Dr. Wasi Haider Butt for their cooperation and prudent suggestions.

My acknowledgement would be incomplete without thanking the biggest source of my strength, my family. I am profusely thankful to my beloved mother and father (late) who raised me when I was not capable of walking and continued to support me throughout every department of my life and my loving brothers who were with me through my thick and thin.

Finally, I would like to express my gratitude to all my friends and the individuals who have encouraged and supported me through this entire period.

*Dedicated to my exceptional parents: **Qaiser Rashid (Late) & Abida Kanwal**, and adored brothers whose tremendous support and cooperation led me to this accomplishment.*



## Abstract

The great Covid-19 pandemic affected billions of people's lives personally and socially. The impact on the public's psychological health were significant as they affected the ways in which people lived, worked, and socialized. It became a hot topic of discussion over the social media platforms as people communicated and expressed their views and detrimental effects on their psychological health. Coronavirus is a new type of infectious disease and to control its rapid spread led to social distancing because of its airborne properties and lack of pharmaceutical measures. Social media is now considered as a main information hub because information is shared over a large scale. People share their emotions and views related to any specific topic through their discussions. The research involves analyzing the people's views and thoughts shared on Twitter platform related to Covid-19 pandemic and its detrimental or non-detrimental effects on public's mental health by using machine learning algorithm. Sentiment analysis is a conventional method to explore people's views by browsing through human-generated textual content from online users. The primary objective of the research is to analyze people's views related to Covid19 pandemic by classifying the Tweets collected from the social platform, Twitter. The accuracy of the classification method is enhanced by using the word embedding approach. Deep learning embedding models like BERT and its variants have been employed to generate high-dimensional word vectors to conserve the semantic information of words. These word vectors are then employed to train the model for the classification of the tweet in five sentiments. As a result, tweets are classified as Positive, Extremely Positive, Negative, Extremely Negative, and Neutral. The methodology is tested on publicly available Tweets dataset on Kaggle, which was split into 90:10 ratio as training and testing sets respectively. The BERT and MiniLM uncased classification models among all the models achieved highest accuracy of 88% and 93% with the Kaggle dataset. This analysis can assist the medial health authorities to monitor health information, conduct, and plan interventions to lower the pandemic effect and can help government to take precautionary measures.

**Keywords:** *sentiment analysis, Covid-19, tweets, coronavirus, Twitter, machine learning algorithm, deep word embedding, BERT.*

# Table of Contents

<b>DECLARATION.....</b>	<b>I</b>
<b>LANGUAGE CORRECTNESS CERTIFICATE.....</b>	<b>II</b>
<b>COPYRIGHT STATEMENT .....</b>	<b>III</b>
<b>ACKNOWLEDGEMENTS .....</b>	<b>IV</b>
<b>ABSTRACT .....</b>	<b>VI</b>
<b>LIST OF FIGURES .....</b>	<b>X</b>
<b>LIST OF TABLES .....</b>	<b>XI</b>
<b>CHAPTER 1: INTRODUCTION.....</b>	<b>1</b>
1.1    MOTIVATION.....	2
1.2    PROBLEM STATEMENT.....	2
1.3    AIMS AND OBJECTIVES .....	3
1.4    STRUCTURE OF THESIS .....	3
<b>CHAPTER 2: SENTIMENT ANALYSIS.....</b>	<b>4</b>
2.1    SENTIMENT ANALYSIS.....	4
2.2    LEVELS OF SENTIMENT ANALYSIS .....	4
2.3    SENTIMENT ANALYSIS APPROACHES .....	5
2.4    SOCIAL MEDIA DATA SENTIMENT ANALYSIS .....	7
2.5    TWITTER SENTIMENT ANALYSIS .....	8
SUMMARY .....	11
<b>CHAPTER 3: LITERATURE REVIEW.....</b>	<b>12</b>
3.1    WORD EMBEDDINGS.....	12
3.1.1 <i>Word2Vec</i> .....	12
3.1.2 <i>BERT</i> .....	20
3.2    RESEARCH GAPS .....	23
3.3    RESEARCH CONTRIBUTIONS .....	24
<b>CHAPTER 4: METHODOLOGY.....</b>	<b>25</b>
4.1    DATA PRE-PROCESSING.....	25
4.1.1 <i>Removing Twitter Links</i> .....	25
4.1.2 <i>Removing Twitter Hashtags</i> .....	25
4.1.3 <i>Removing Twitter Handles (@user)</i> .....	26
4.1.4 <i>Removing Punctuations, Numbers, and Special Characters</i> .....	26

4.1.5	<i>Removing White Spaces</i> .....	26
4.1.6	<i>Removing Short Tweets</i> .....	26
4.2	FEATURE SELECTION .....	28
4.2.1	<i>Word Embedding</i> .....	28
4.2.2	<i>Word2Vec</i> .....	29
4.3	ALGORITHM .....	33
4.3.1	<i>Machine Learning (ML) Model</i> .....	33
4.3.1.1	<i>Random Forest</i> .....	33
4.3.2	<i>Deep Learning (DL) Models</i> .....	34
4.3.2.1	<i>Bidirectional Encoder Representations from Transformers</i> .....	34
4.3.2.1.1	<i>Model Overview</i> .....	35
4.3.2.1.2	<i>Input Output Format</i> .....	35
4.3.2.1.3	<i>BERT Pre-Training</i> .....	37
4.3.2.1.4	<i>BERT Fine Tuning</i> .....	38
4.3.2.2	<i>Distil BERT</i> .....	40
4.3.2.3	<i>DeBERTa-v3-base</i> .....	41
4.3.2.4	<i>MiniLM-L12-H384-uncased</i> .....	42
4.3.2.5	<i>Twitter RoBERTa Base</i> .....	42
4.3.2.6	<i>Twitter RoBERTa Base Sentiment</i> .....	43
4.3.2.7	<i>Twitter RoBERTa Base 2022-154M</i> .....	43
	SUMMARY .....	44
<b>CHAPTER 5: EXPERIMENT &amp; RESULTS</b> .....		<b>45</b>
5.1	DATASET .....	45
5.1.1	<i>Realtime Twitter Dataset</i> .....	45
5.1.2	<i>Kaggle Corona NLP - Text Classification Dataset</i> .....	46
5.1.3	<i>Train-Test Split</i> .....	49
5.2	PERFORMANCE METRICS .....	51
5.2.1	<i>Accuracy/ Precision</i> .....	51
5.2.2	<i>Recall</i> .....	51
5.2.3	<i>F1-score</i> .....	51
5.2.4	<i>Confusion Matrix</i> .....	51
5.2.5	<i>Experimental Setup &amp; Parameters involved in Models Training</i> .....	52
5.3	EXPERIMENTAL RESULTS OF BERT WITH MULTIPLE HYPERPARAMETERS .....	53
5.3.1	<i>Effects of Batch Size Number on the Accuracy of BERT</i> .....	53
5.3.2	<i>Effects of Epochs Number on the Accuracy of BERT</i> .....	53
5.3.3	<i>Effects of Learning Rate on the Accuracy of BERT</i> .....	54

5.4 EXPERIMENTAL RESULTS OF BERT .....	55
5.5 COMPARISON OF THE MODELS ON TWITTER SENTIMENT ANALYSIS .....	56
5.6 COMPARISON WITH EXISTING TECHNIQUES .....	57
SUMMARY .....	57
<b>CHAPTER 6: CONCLUSION.....</b>	<b>58</b>
<b>REFERENCES.....</b>	<b>59</b>

## List of Figures

<b>Figure 1: Increase in Twitter Users</b> .....	9
<b>Figure 2: Why People Use Twitter</b> .....	9
<b>Figure 3: Overview of Steps involved in Preprocessing.</b> .....	27
<b>Figure 4: Maximum Sentence Tokens in Dataset</b> .....	27
<b>Figure 5: CBOW Model Architecture</b> .....	30
<b>Figure 6: Skip-gram Model Architecture</b> .....	31
<b>Figure 7: 200-Dimensional Word Vector Representation for Word “Food”</b> .....	32
<b>Figure 8: Proposed RF Model for Twitter Sentiment Classification</b> .....	34
<b>Figure 9: BERT Tokenization and Vectorization</b> .....	37
<b>Figure 10: Flow Chart of Proposed BERT Model</b> .....	40
<b>Figure 11: Class Distribution by Label</b> .....	47
<b>Figure 12: Word Cloud of Dataset</b> .....	47
<b>Figure 13: Confusion Matrix of the Proposed BERT Test Results</b> .....	52

## List of Tables

<b>Table 1: Summary of Existing ML Sentiment Analysis models</b> .....	19
<b>Table 2: Summary of Existing Sentiment Analysis models using BERT.</b> .....	23
<b>Table 3: Comparison of Raw and Processed Tweets</b> .....	28
<b>Table 4: Most similar words for the target word “Dinner” Generated by Word2Vec</b> ...	32
<b>Table 5: Splitting of Train-Val Sets for Each Sentiment Class</b> .....	39
<b>Table 6: Randomly Selected Tweets from Real Time Twitter Dataset</b> .....	45
<b>Table 7: List of Dataset Attributes</b> .....	46
<b>Table 8: Randomly Selected Tweets from Kaggle Dataset for Each Class</b> .....	48
<b>Table 9: Train Dataset Count for Each Class</b> .....	50
<b>Table 10: Test Dataset Count for Each Class</b> .....	50
<b>Table 11: Fine-tuned Hyperparameters Involved in Models’ Training</b> .....	52
<b>Table 12: Effects of Batch Size Number on the Accuracy of BERT</b> .....	53
<b>Table 13: Effects of Epochs Number on the Accuracy of BERT</b> .....	54
<b>Table 14: Effects of Learning Rate on the Accuracy of BERT</b> .....	54
<b>Table 15: Experimental Results of BERT model on Test Dataset</b> .....	55
<b>Table 16: Experimental Results of BERT model on Custom Twitter Dataset</b> .....	56
<b>Table 17: Comparison of the Models on Twitter Sentiment Analysis</b> .....	57
<b>Table 18: Comparison of the Model with Existing Techniques</b> .....	57

## CHAPTER 1: INTRODUCTION

A new virus surfaced in Wuhan, the city of China in December 2019 which was later called as Covid-19 virus. The virus was spread throughout the whole world except Antarctica continent in the early 2020, causing a huge number of deaths because of its transmittable characteristic and causing common infections, but no medical treatment was introduced against it. The Covid-19 pandemic was labelled as the most significant worldwide disaster since the World Wars. To control the spread of Covid-19 virus, non-pharmaceutical measures such as emergency lockdowns, wearing masks in public places, isolation, quarantine, and social distancing were introduced. These precautionary strategies on one hand helped in the reduction of infection spread but on the other hand they caused negative impact on the mental health of public. Several studies conducted on Covid-19 virus on the global level showed detrimental impact on the public's psychological health [12]. The major pandemic, which economically and socially affected billions of lives, motivated the scientific community to come up with solutions based on computer-aided digital technologies for the diagnosis, prevention, and estimation of Covid19. Some of these efforts focus on statistical and AI-based analysis of available Covid-19 data.

Considering the first line of defense measures, the majority of the public turned to social media to express their views concerning the pandemic happening in the globe. In this modern technological era, the social media's impact has become more noticeable than ever before. The usage and investment of time by people in social media applications has thus turned these sites into the global big data center. The effectiveness of these social media platforms stems from their ability to highlight valuable insights from multiple perspectives on events in real time. In addition, social platforms serve as rapidly expanding social information structures that have substantial influence.

The most used social media site is the most prevalent social networking platform. The most used tool by millions of people to express their views to others across the world is microblogging. Twitter, a widely used microblogging platform, allows users to express their thoughts through short texts known as microblogs, which are limited to 140 characters [1]. With approximately 326 million monthly active users, Twitter is accessible via SMS, mobile devices and the website. Notably, 80% of current Twitter users interact with the platform via mobile phones. These microblogs, combined with user-related information, collectively referred to as tweet objects, provide researchers with a valuable source of data. Once the data

is processed, it can be subjected to various analyzes to gain insight into public opinions. Numerous studies have demonstrated the effectiveness of Twitter data analysis, particularly in areas like election polling, stock market, forecasting, crime analysis, and disaster management [13]. Analyzing tweets during and after the Covid-19 pandemic is of significant value as public responses and conditions evolve rapidly during this critical period. This study aims to explore the changing emotions and concerns of individuals regarding Covid-19, starting from the initial stages of the pandemic to the present.

## **1.1 Motivation**

People across the world got concerned about the disease, with the increase in the severity of the Covid-19 pandemic. Following the precautionary measures, people shared their views via blogs, messages, comments, etc. Analyzing the content of the shared posts can assist the government to recognize the public's basic needs, their interests and can also be supportive in improving the measures taken by various organizations. Tracking and analyzing tweets is a rational approach of digging into people's opinions, views, behaviors, and responses regarding Covid19 pandemic since their attitudes changed constantly in various situations. In addition, tweets show real-time responses from many people to better understand the change in sentimental trend during the disease pandemic.

## **1.2 Problem Statement**

The current analyzing tweet polarity methods mostly are lexicon-based or machine-based, but these methods don't conserve the contextual semantics of words in the text. Many learning-based approaches focus on producing functional features while some researchers use word-based features, such as Term Frequency-Inverse Document Frequency (TF-IDF) and classified the tweets in three main sentiments. Many of the learning strategies available on Twitter's emotional analysis focus on feature engineering. Currently, researchers have introduced the concept of deep learning method [1] which is the advanced approach for sentiment analysis and provides better performance results by embedding semantic text information into text. The research aims to capture the sentiment of the tweets through a word embedding approach which conserves the semantics of the words. As a result, the tweets will be classified in five different sentiments i.e., Positive, Extremely Positive, Negative, Extremely Negative and Neutral. Also, this will help in visualizing the results to see the people's reactions on Twitter.



### 1.3 Aims and Objectives

The primary goals of this research are as follows:

- Review and comparison of recent developments in Covid-19 sentiment analysis
- Precise sentiment analysis of Covid-19 tweets from various time intervals using deep word embedding by conserving the semantics of the words.
- Classification of Twitter tweets as Positive, Extremely Positive, Neutral, Negative and Extremely Negative using deep learning
- Collection of novel Twitter data set related to Covid-19 from Twitter platform.
- A detailed ablation study to experiment with different available models.
- Analyzing public's reactions through Story Generation and Visualization from Covid Tweets.

### 1.4 Structure of Thesis

This research work is planned as follows:

**Chapter 2** covers the significance of sentiment analysis.

**Chapter 3** gives the literature review and the major related work done by the various researchers for Covid-19 sentiment analysis in the past few years.

**Chapter 4** consists of the proposed methodology in detail. It includes two frameworks; the first framework covers the machine learning model, and the other framework covers the deep learning models.

**Chapter 5** presents the databases used for evaluation. Detailed discussions of all experimental results are provided along with relevant figures and tables.

**Chapter 6** serves as the conclusion of the paper and outlines future prospects and potential avenues for further research.

## CHAPTER 2: SENTIMENT ANALYSIS

### 2.1 Sentiment Analysis

Sentiment analysis is a collection of semantic operations within the field of natural language processing that are performed automatically [14]. The primary purpose of sentiment analysis is to analyze the polarity of people's comments or opinions pertaining to a specific topic and classify them as either positive or negative [7]. Sentiment analysis has many benefits in various areas like in industries sentiment analysis can be used to get product's feedbacks through which corporations can understand users' view regarding the product's quality and offered services to enhance the customer satisfaction and experience. The review posts on social media can help to evaluate the client's review. The traditional method of sentiment analysis is carried out by casting some ideas or a list of product-based client assessment questions and text-based review results with numbers, points, and reviews. Although these results provide guidance and assistance in managing efforts towards the most important changes to be made, investigating the text manually is a tedious task and almost difficult to be done. Without a fixed professional cycle, organizations will remain at a high level in secrecy about how to make their progress.

### 2.2 Levels of Sentiment Analysis

Sentiment analysis can be divided into three primary levels depending on the scope of the analysis: Document Level, Sentence Level, and Entity or Aspect-Level as mentioned in [41].

- **Document Level:** The goal of this level is to categorize and classify all the sentiments expressed within the document; this level specifically works in the uniqueness of the document because multiple topics cannot be edited because the paper at this stage of analysis should be about the same thing.
- **Sentence level:** This level is used to classify a sentence, for example to ensure that whether a statement is negative or positive, at this level may or may not have an impact on the class of a neutral sentence.
- **Entity or Aspect Level:** This level is also known as aspect level because at this stage each entity of the sentence is briefly focused to recognize any type of mentioned entities that attract feelings like people, places and products etc. and understand, what type is negative or positive or both aspects.

There are different methods of performing sentiment analysis depending on the type, nature and domain of the text as well as the potential applications. Sentiment analysis is commonly categorized into two main groups [41]: language processing-based sentiment analysis and application-oriented sentiment analysis.

- **Language Processing-Based Sentiment Analysis:** This approach utilizes sentiment dictionaries, also known as lexicons, to analyze sentiment. These dictionaries leverage grammatical structures, language norms, and semantics to accurately classify the sentiment of a sentence into positive, negative, or neutral categories. Domain-specific linguistic dictionaries or corpora can be utilized to construct these lexicons. Dictionary-based approaches tend to be more comprehensive and intricate due to their bootstrapping nature, whereas corpus-based approaches have limitations and are not easily transferable to different domains. It has been observed that sentiment dictionaries enhance the accuracy of polarity and subjectivity classification for sentences in any given text.
- **Application-Oriented Sentiment Analysis:** Application-oriented sentiment analysis involves analyzing the sentiment of textual data in specific applications or domains, such as product reviews, social media posts, news articles, or customer feedback. Application-oriented sentiment analysis can help businesses and organizations understand the sentiment of their customers and target audiences in a specific domain, identify areas for improvement, and make data-driven decisions based on sentiment analysis results. Various tools can be used to perform application-oriented sentiment analysis, and machine learning algorithms such as SVM, Naive Bayes, Maximum Entropy, and others are commonly used.

## 2.3 Sentiment Analysis Approaches

There are mainly three approaches for the classification of sentiments i.e., machine learning based approaches, lexical based approaches and linguistic analysis [41].

### I. Machine Learning based Methods:

Sentiment analysis through supervised machine learning based methods involves training a model with labeled dataset to classify the sentiment of a given text. The labeled dataset consists of textual data with known sentiments like positive, negative, or neutral. In training the model, the textual data is represented as feature vectors like unigrams, bigrams or trigrams. Unigrams represent single words, bigrams represent

two consecutive word phrases, and trigrams represent three consecutive word phrases. Higher order n-grams represent more complex cases. For example, the sentence "That's not good." can be classified as positive using the unigram approach due to the presence of the word "good". While using the bigram approach, the phrase "not good" can be identified as a negative sentiment classification. The classification of sentiments using supervised machine learning typically involves models such as Support Vector Machines (SVM), Naive Bayes, and Random Forest. These models are trained on a labeled dataset and learn to recognize patterns that can predict the sentiment of new text. The studies [40] showed that these approaches achieve accuracy of 60 to 80% depending on the model's training and the size of the corpus data.

## **II. Lexical based Methods:**

A lexical-based approach identifies the sentiment or emotion expressed in the given text using a lexicon or dictionary containing polar or opinion terms. This approach relies on detecting words that indicate a particular sentiment or emotion and analyzing their polarity to determine the overall sentiment of the text. The method involves creating a lexicon that provides a sentiment score for each word or phrase based on its polarity, whether positive, negative or neutral. Multiple techniques such as manual annotation by human raters, unsupervised learning, or supervised machine learning can be used to determine sentiment scores. After the lexicon is determined, the sentiment lexicon-based method involves dividing the input text into tokens, such as words or phrases, and comparing each token to the contents of the lexicon. If a match is identified, the sentiment score assigned to the token is used to determine the overall sentiment of the text. Multiple approaches can be used to calculate a sentiment score, including simply adding up the number of positive and negative words in a text, a weighted number based on the strength of sentiment expressed by each word, or a more advanced algorithm that considers the context in which the words appear [2].

## **III. Linguistic Method:**

The sentiment analysis via linguistic approach involves analyzing the language and grammar used in a text to identify its sentiment. This approach depends on the identification and analysis of the presence of specific words, phrases and grammatical structures that are linked with positive or negative sentiments. For example, words like "love," "happy," or "awesome" specify positive sentiment, while words like

"hate," "sad," or "terrible" specify negative sentiment. Language structures such as negation (e.g., "not good") or intensifiers (e.g., "very good") also affect the sentiment of the text. Linguistic methods can be rule-based (hand-made rules) or use machine learning techniques (learn patterns through training) to identify sentiment-carrying words and structures. But these pre-trained rules or lexicons cannot capture complex feelings.

All these three discussed approaches can be used as a stand-alone method or can be used as a combination of methods.

## **2.4 Social Media Data Sentiment Analysis**

In this advanced technological world where communication technology has enhanced and approach to social media got better, people express their views or opinions freely on social media sites like Facebook, Instagram, Twitter, YouTube etc. Increasingly, a lot of data is generated through many websites and online web-based media, and we can gain more knowledge and information by exploring and classifying this data. For instance, in mining web news "Big Data" and an automated mining-based system can help in tracking, analyzing, and classifying the daily news into categories and help editors in effectively managing the news stories [18].

Social media networks have provided an easy and accessible approach to data. Since all the information provided by social media isn't important, choosing the right data sources and domains from a wide range of data can be a major challenge in any research field. Six various forms of Internet-based platforms have been discussed in "Mastering social media mining with R" book [17].

- i. Networking forums: these services allow users to connect with other users with similar interests in a domain, for example LinkedIn and Facebook.
- ii. Micro-blogging forums: these services allow users to share short updates that are pushed for fans and subscribers, like Twitter and Tumblr.
- iii. Photo-sharing forums: resources like Instagram and Flickr allow users to share their self-created photos.
- iv. Video sharing forums: services like YouTube and Vimeo allow users to share their own personalized videos.
- v. Stack exchange forums: online forums like GitHub allow users to share concepts and carry-on discussions related to the posts.

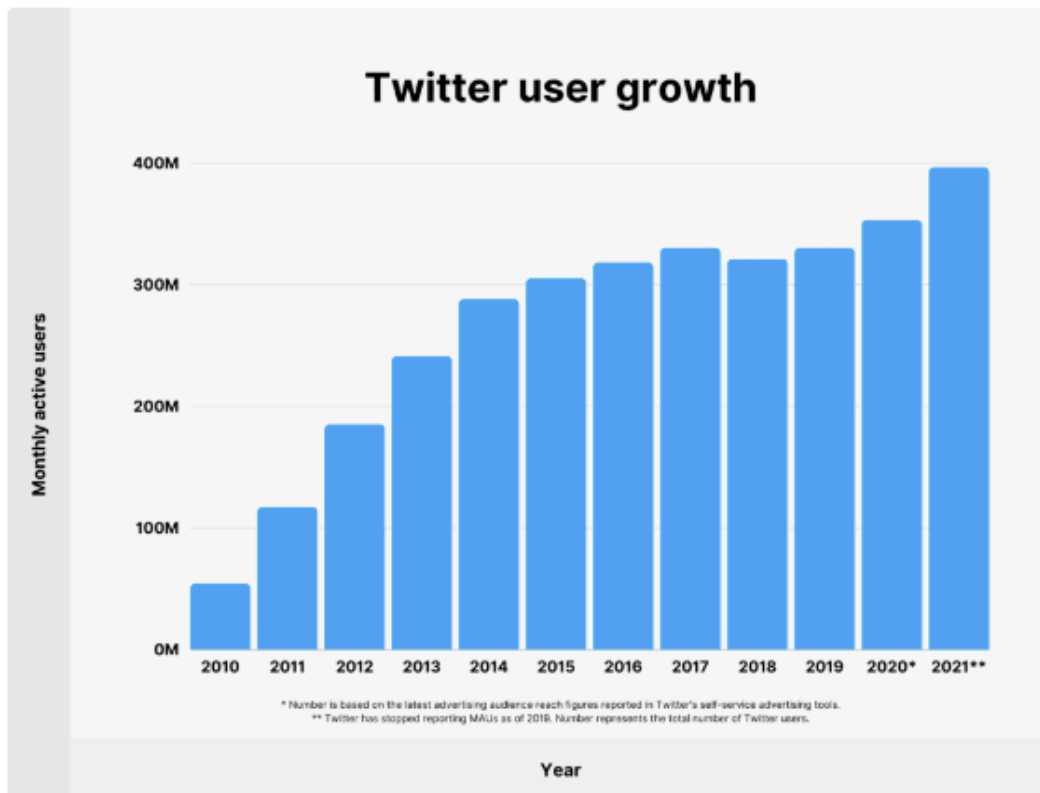
- vi. Instant messaging forums: services like WhatsApp and WeChat let users connect and communicate directly with their relatives and friends.

The increasing number of posts on different social media platforms has made sentiment and opinion mining as one of the important research areas. Extracting people's view from their posts on social media is quite a challenging task. The aim of this research is to analyze the people's sentiments concerning Covid-19, belonging to the virus family, which has infected humans and animals in vast range. Today, to analyze the virus sentiments, data has been collected during the past two years but getting insight into this large range data is not trivial. There are some works that has been done on Facebook sentimental analysis, however this research focuses on Twitter sentimental analysis.

## **2.5 Twitter Sentiment Analysis**

Social media is a rich platform to gain insight about people's views and sentiment related to different topics as this platform allows them to communicate and share their views freely on daily basis through Facebook, Twitter, or Instagram, etc. Thus, a microblog is a platform where users are allowed to share brief messages, other websites' links, pictures, or videos. Usually, a person writes and shares a message on microblog and hundreds of thousands of people, known as followers, read that message. Generally, microblog is updated by the user personally except if it represents any corporate profiles or political parties which are updated by group of executives. Microblogging users typically update their blogs regularly, whereas the most active users update it each hour to provide a timeline of interesting information to their followers. As with blogs, microblogs can cover various distinct topics, some very private and shared with a very small group of followers, and others where appealing information is provided to a large group of followers. Conversely, artists and celebrities have their profiles whose estimation is more about popularity than the news provided through their account of the microblogging service.

On 16 July 2006, the first initial post on the microblogging Twitter service took place and ever since then its popularity has been increased such that it is even deemed as a subject of study in various scientific fields. Twitter, being the eighth most desired website in the universe, shows the significance of this platform. Twitter has an average of almost 11 million posts each day [19]. According to the statistics in [4], Twitter is the most used social platform and has around 400 million active accounts.



**Figure 1: Increase in Twitter Users**

Twitter is mainly used as a source of news in the whole world. Other commonly cited reasons for its usage include amusement, following business accounts or keeping in touch with friends and family. The following figure 2 shows a comprehensive list of reasons for Twitter usage by people [19].

Reason	Share of respondents
To get news	48%
To get entertainment	48%
To keep contact with friends and family	34%
To follow brands / companies	33%
To strengthen professional network	14%
Other purposes	12%

**Source:** Statista

**Figure 2: Why People Use Twitter**

Nowadays, sentiment-aware systems find applications in various research fields, ranging from business to social sciences [15]. Micro-blogging platforms, like Twitter, provide individuals with an accessible means to share their views worldwide, thanks to easy access, real-time responses, regular posts, and minimal production time. Researchers consider Twitter a valuable and reliable source of data. However, the nature of tweets, which consist of short texts with diverse words and abbreviations, makes it challenging to extract sentiments using traditional Natural Language Processing systems. To overcome this challenge, researchers have turned to deep learning and advanced machine learning techniques to extract and analyze the polarity of the text. Common abbreviations such as Fb/FB for Facebook, Insta for Instagram, B4 for before, and many others are frequently used in tweets, further complicating sentiment analysis [16].

The brevity of tweets, limited to 140 characters, poses another constraint and prompts people to use unconventional words and phrases that may not be included in traditional language processing [1]. Nonetheless, Twitter remains a valuable resource for gaining insights into public opinions and analyzing sentiments, with each tweet carrying significance in determining its positive, negative, or neutral sentiment. Recently, the text limitation has been increased to 280 characters per tweet.

Twitter is the platform that broadcasts all types of information and propagates people's views on various topics of interests like political current affairs, economics, industry, and many more. On the regular basis, Twitter users post their opinions on any selective news article like newly purchased product or political event, or eventually related to everything that is happening in the world around them. This piqued the NLP research community's interest, which commenced them to study the blogs published on Twitter.

Facebook, being the largest social network globally, has a vast user base. However, it is not commonly utilized for sentiment analysis due to the unstructured nature of its data. Users often employ abbreviations and make spelling mistakes in their posts, making the analysis of such complex data challenging. In contrast, platforms like Twitter and Facebook are frequently utilized to gather user reviews, page loads, status updates, and comments [22]. Data for the study was collected from various social media sources, including forums, blogs, blog spot, Expedia, WordPress, mainstream media, aggregators, YouTube, Twitter, and Facebook. The findings revealed that 88% of the collected data originated from Twitter. Any other social media source isn't preferred because of the limitation in the number of reviews or data that can be extracted, for example in YouTube, Blogspot, and WordPress.



Many recent researched have used Twitter as a source for sentimental analysis through a variety of application forms, such as political predictions, to inspect the efficacy of a service or policy, and to keep track of contagious disease and communal health problem. For example, when Ebola virus broke out, Fung et al. [20] scientifically reviewed twelve existing studies associated with Ebola virus and social media in a cross-sectional manner and demonstrated the effectiveness of using electronic sentiment analysis in the community health field. 7 articles out of 12 were Twitter research, which also revealed the sentiment analysis trends and highlighted the preference for Twitter over other social media platforms among researchers related to sentimental analysis issues. Rasool et al. [21] conducted apparel brands research by implementing various sentiment analysis models to evaluate the public views shared on Twitter for two leading international apparel brands, i.e., Adidas and Nike. The positive and negative opinions of online users towards both brands were compared. The results showed that positive reviews of Adidas were more than Nike, which concluded the popularity of Adidas brand among online viewers. It has also been found that users compare other related brands online while making a brand decision.

## **Summary**

The section explained the significance of sentiment analysis, which is a set of semantic operations to analyze the polarity of a text related to a specific topic and classify the underlying text as positive or negative. Sentiment analysis on social media platforms such as Twitter, a major source of communication, is increasingly valuable for understanding public sentiment and emotions during events such as news, daily events, pandemics, and natural disasters. Research shows that conducting a comprehensive analysis of social media content can provide valuable insights into predicting sentiments and panic levels during a corona outbreak, as well as assessing the psychological impact on individuals.

## CHAPTER 3: LITERATURE REVIEW

Several research works have been done by different researchers related to Twitter sentiment analysis and some studies are related to the Covid-19 sentiment analysis.

### 3.1 Word Embeddings

#### 3.1.1 Word2Vec

In a study conducted by Aditya Sharma and Alex Daniels, sentiment analysis of real-time Twitter data for the 2019 elections was performed using the word2vec feature selection model and a random forest machine learning algorithm for sentiment classification [1]. The data for the study consisted of Twitter election-related posts from India, collected through the Twitter developer API using hashtags associated with Indian politics in 2019, excluding retweets. The collected tweets were categorized into positive and negative sentiments. Pre-processing steps were applied, including the removal of Twitter handles, numbers, punctuation, special characters, and insignificant words. The data was then normalized. Word2vec was used to generate word vectors, and by taking the mean of the word vectors in each tweet, a vector representation of the entire tweet was created, with a fixed vector length of 200. The same process was repeated to obtain vectors for the test data. The feature sets derived from word2vec were used to train a random forest model for sentiment analysis. The training dataset consisted of 18,685 tweets, with 12,890 positive and 5,795 negative tweets. The test dataset was used to evaluate the model's performance. F1 score was used as an evaluation metric. The word2vec feature selection model was compared to traditional methods such as Bag-of-Words (BOW) and TF-IDF, and it exhibited significantly higher accuracy, achieving 86.87% accuracy compared to BOW and TF-IDF. The contextual semantics captured by word2vec improved the quality of features, leading to enhanced accuracy in sentiment analysis. In a study conducted by Rezaeinia et al. [2], a new approach called Improved Word Vector (IWV) was proposed to enhance the precision of pre-trained word embedding in analyzing sentiments. The approach incorporated Speech Recognition (POS) tags, lexicon-based methods, and Word2Vec/GloVe methods. The accuracy of the proposed approach was evaluated using various deep learning models and sentiment datasets. Experimental results demonstrated the effectiveness of the Improved Word Vector (IWV) in sentiment analysis, yielding highly efficient sentiment analysis outcomes. The proposed algorithm detected a sentence and returned the vectors of the improved wording of the

sentence. In the first step, a fixed vector was assigned to each POS mark. In the second step, the vector of each input phrase was extracted from Word2Vec or GloVe data sets and if the word was not found in the datasets, then its vector was generated randomly. In the third step, the POS tag for each word was determined and assigned a fixed vector to each one. The next step involved the extraction of sentiment scores for every word from all the lexicons and normalized them. If a word doesn't appear in any of the lexicons, its score was zero. The vectors generated in each step were linked to other vectors from the previous steps. The authors developed a deep learning model (known as Model 1) to test the generated vectors on known databases. The model consisted of three CNNs, a pooling, and a fully linked layer and improved word vectors (IWV) were the inputs to the model. The accuracy of the proposed method was ensured by testing it with different deep learning models and sentiment datasets. In a study conducted by B. Oscar et al. [3], the authors explored methods for emotional analysis of tweets related to the establishment of the U.S. Army base in Ghana. The collected tweets underwent pre-processing steps such as the removal of stop words, tokenization, and word stemming to prepare the data for feature extraction. The Word2Vec Skip-Gram model was chosen for its ability to preserve context while maintaining accuracy. Various parameters such as the training algorithm, size, context window, and sub-sampling were carefully considered. Negative sampling was used as the training algorithm, which was found to be mathematically efficient compared to hierarchical softmax. The neural network's hidden layer was set to 300 dimensions, resulting in improved word embedding. A fixed context window of 10 was utilized for the skip-gram model. To address the imbalance between rare and common words in the database,  $1e-3$  sub-sampling rate was applied. The word embedding model produced word vectors of size  $m * n$ , where  $m$  represents the dictionary size and  $n$  denotes the size of the hidden layer. The word vectors were split into a training set (70%) and a test set (30%). VADER (Valence Aware Dictionary for sentiment Reasoning), an emotional analysis engine, was used to determine the sentiment variability of different tweets. The resulting variables were assigned to 70% of the relevant word vectors as the training data set. A random forest model consisting of 100 decision trees was trained using the labeled vectors. The trained classifier model was then used to predict the polarity of the experimental database. The performance of the Random Forest classifier was evaluated by calculating accuracy, recall, and F1-score. The model achieved an overall accuracy of 81% in predicting sentiment polarities, which was attributed to the quality of the word vectors produced by the skip-gram model. An Indonesian word embedding model was designed and

used to analyze sentiments by Farhan and Khodra [4]. The algorithm proposed in this study builds upon Collobert's C&W embedded model, which captures the contextual information of word formation. By incorporating sentiment word embedding, the F1-score of sentiment classification for Twitter data was improved. The words were embedded with sentiment labels of positive, negative, and neutral to enhance the overall performance of the classification task. The data collected from the TripAdvisor website explicitly was used to create the embedding of the words. The reviews related to Indonesian restaurants and hotels were collected. The final database contained 306,448 reviews/opinions based on user feedback and ratings assigned to each review. The data was pre-processed using formalization (INANLP, Indonesia NLP pipeline), removing unnecessary punctuation marks, combining numbers and icons, case folding and using regular expressions. Pre-processed data was then used to train the embedding algorithm. The DeepNL algorithm was used to produce the first model. Next, a modified Word2Vec model was employed using the same corpus, incorporating sentimental labels for each review, to generate sentiment embeddings. These sentiment embeddings were then integrated into the review database as additional features. The resulting dataset, comprising embedded features, was used to train the selected classification models, namely random forest, SVM, and neural processing networks. The sentiment classification function was based on the embedding of specific words, along with other commonly used feature representations like bag of words, TF-IDF, and standard word embeddings. The trained classification model was evaluated using both 10-fold cross-validation and a separate test set. The performance metric used for comparison was the macro F1-score, computed for each sentiment label. The proposed sentiment-specific embedded models demonstrated higher accuracy compared to the original Word2Vec embedding model. Specifically, the SVM model achieved an F1-score of 0.837, surpassing the F1-scores of other embedding models such as Word2Vec (0.7489) and C&W (0.7321).

Imaduddin et al. used hotel reviews data to perform sentimental analysis found on the Traveloka website [5]. In this research, an emotional analysis system was developed using the Deep Learning approach. The authors crawled hotel review data using the WebHarvy tool to collect data. The collected data was first cleaned to produce high classification results. After the pre-processing, the data was manually marked as sentiment positive and negative. In this paper the authors used the corporate data component provided by Indonesia's Wikipedia. For data analysis, word2vec and doc2vec models were created using the Python genism library. The parameters of the training algorithm, size (set to 300) and context (set to

10) were carefully chosen to improve word embedding. Both word2vec and doc2vec models were trained on preprocessed data and saved in a suitable data file format. The word vectors generated by the word embedding model were of dimensions  $m * n$ , where  $m$  represents the size of the dictionary and  $n$  represents the size of the hidden layer. The same parameter settings were used when building the glove model. LSTM (Long Short-Term Memory), a type of Deep Learning algorithm, was used for sentiment classification. It was used to determine the polarity of each tweet in the dataset. Hotel review data were preprocessed and converted into nested vectors, each input consisting of 50 words. The output data was classified into two polarities: positive and negative. The performance of the classification model was evaluated using 10-fold cross-validation. The results showed that the glove model achieved the highest accuracy of 95.52% among the different models. Other models such as Word2vec CBOW achieved 92.72% accuracy, Word2vec skip-gram achieved 91.81% accuracy and Doc2vec achieved 94.81% accuracy. The Glove's superior performance can be attributed to its combination of skip-gram and CBOW models.

In their article [6], Amin et al. presented a novel approach for sentiment classification of Bengali comments using word2vec and sentiment extraction from words. By combining the word co-occurrence score from word2vec with the sentiment polarity score of words, they achieved an accuracy of 75.5%. The study focused on creating a dataset of one-line and multi-line comments from a Bengali microblogging website, where each comment was labeled as positive or negative based on public opinion. The size of the database and the classification accuracy were found to be directly related, indicating the importance of word2vec word embedding. Data cleaning was performed by removing unnecessary spaces, punctuation, and unknown characters from the datasets. Over 16,000 Bengali comments were collected from popular blogging websites, and each comment was marked as positive or negative based on surveys conducted to gather public views. The dataset was split into two sets, one containing positive comments and the other containing negative comments. Initially, the accuracy of the model was not satisfactory compared to existing sentiment analysis methods, attributed to the dataset volume and the sentiment attributes of words. To overcome these challenges, the number of sentences in the database was increased, and the emotional aspects of words were taken into consideration. A significant improvement was introduced by creating a list of highly favorable and extremely negative words, particularly adjectives, with each word assigned a polarity value ranging from -1 to +1 based on its positivity or negativity. These polarity scores were calculated by considering the frequency of the words

in the comments. The authors trained the model using 90% of randomly selected comments, reserving the remaining 10% for evaluation. The proposed model underwent six iterations, with each iteration training on an additional 2,500 comments from the training database. The results showed that as the database size increased, the model's accuracy improved. When trained on a total of 15,000 comments, the model achieved an accuracy of 75.5%.

Yue and Li [7] proposed a mixed Word2vec-CNN-BiLSTM model that combines Word Vector Model (Word2vec), Bidirectional Long-term and Short-term Memory network (BiLSTM), and Convolutional Neural Network (CNN) for sentiment classification. The model achieved an accuracy of 91.48%. They used Quora's Internet dataset for text classification and compared the strengths and weaknesses of LSTM and CNN. The recommended design of the model consisted of three sections: Pre-processing, Convolution, and BiLSTM/Fully Connected Section. In the pre-processing stage, the data was cleaned, and short text data was pre-processed. They also eliminated stop words and low-frequency words from the data. Word2Vec embedding was used to obtain vector representations of the text words. The resulting output vector was then passed as input to the next phase. The convolution and max pooling layers were utilized to extract high-level features from the data. The output data from these layers were fed as input to the BiLSTM network layer. The BiLSTM and fully connected layers were responsible for classifying the sentiment of the document, determining whether it was positive, negative, or neutral. Compared to existing baselines, the proposed model required more training data and training time. However, the results demonstrated that the word2vec-CNN-BiLSTM model could more accurately analyze text, enhancing the accuracy of short text classification.

In the research article [8], a method to enhance the performance of sentiment classification was proposed by combining SSWE (Sentiment-Specific Word Embedding) with a weighted text element (WTFM) model. The WTFM model generated two types of features: denial features based on opposing words in the tweet, and features that matched the tweet with each of the three polarity types using cosine similarity and TF.IDF. These values were used as features and input to the selected classification model. In the proposed SSWE + WTFM model, four WTFM-generated features were combined with SSWE embedding to represent the tweet. The WTFM model was simple yet effective in generating tweet features compared to other methods, and it did not require any external source. Experimental results demonstrated that this approach outperformed two contemporary sentiment classification models: the SSWE model and the National Research Council Canada (NRC) model. The

SSWE model used in the research was trained on a large dataset of well-labeled tweets, consisting of both positive and negative sentiments. The tweet text underwent pre-processing steps, such as URL and expression removal, elimination of special characters (except hashtags, emoticons, question marks, and exclamation marks), conversion of dates into symbols, replacement of ratings with special marks, normalization of numbers and decimals, and removal of negative words already used in the negative elements. The tf.idf values were calculated for each term in the polarity group, treating each group as a document, and tf was normalized by the size of its group. The cosine similarity between the tweet and each sentiment type was computed. Various classification algorithms were tested to determine the most effective ones for the proposed models. The results indicated that the LibLinear and SMO models performed well. Model evaluation was based on accuracy, recall, and F-measure for the positive, neutral, and negative classes. The proposed approach demonstrated superior performance compared to the NRC, which served as the benchmark. The model achieved an F-score of 66.8, outperforming the other models.

The research in [9] provided comparative research for different models such as skip gram and Continuous Bag of Words. The authors implemented Glove model and offered its possible use in sentimental analysis. The word vectors created using the Glove method were supplied to RNN. The dot product of word vectors was calculated to find the similarity between two words. Highly similar words were found by taking the dot product of the vector with all the other word vectors and the words with the vectors that produced the maximum dot product were considered the same. The data was fed to model for sentimental analysis, GloVe then generated vectors for every word (here dimension = 50), then the sentence was converted to the corresponding word vectors and a 3-dimensional vector was given to Recurrent neural network layer (RNN). The RNN output label categories were 2 polarities (positive and negative). The model was trained to provide analysis of the sentiment of the subject. If January was the word given to the model, then Glove showed the same words i.e., the months of the year. Ren et al. [10] proposed to include topic information in word embedding to analyze Twitter sentiment and used a recursive autoencoder to achieve the goal. First, the tweet subject information was created using the Latent Dirichlet Allocation. Second, the existing repetitive autoencoder has been expanded to successfully integrate topic information into their intended function. The researchers investigated the topic enhanced embedding of Twitter tweets classification in supervised learning structure. The recommended method achieved F-measure 78.57% in predicting positive or negative polarity tweets using only

topic enhanced word embedding as features. After combining advanced embedding features and pre-designed handwriting features, performance was improved by 81.02% in macro-F-measure. Database test results showed that improved embedding is more effective in classifying Twitter sentiments.

Sitaula et al. [11] conducted an analysis of individual opinions based on tweets collected from Twitter in Nepal. Researchers have proposed three different methods for extracting features to represent tweets: fast text (ft), domain-specific (ds), and domain-agnostic (da). The ds and da methods were new approaches introduced in the study. Then, three different CNN models were proposed for tweet sentiment classification using ft, ds, and da feature extraction methods. In addition, an integrated CNN model was designed to combine the three CNN models to achieve better results. These CNN models have demonstrated consistent and robust performance. To evaluate the recommended feature extraction methods and CNN models, the authors created a Nepali Twitter sentiment database named NepCOVID19Tweets. This database contained three classes of sentiment: positive, neutral, and negative. Tweets were collected between February 11, 2020 and January 10, 2021 using geolocation filters specific to Nepal. Tweets were searched on Twitter using the keyword #COVID-19 in the Nepali language. Each tweet was pre-processed and sentiment-annotated using a majority voting method. Popular evaluation metrics such as Precision, Recall, F1-score and Accuracy were calculated to assess the performance of the models. Sentiment classification of tweets related to COVID-19 involved three separate steps: embedding vector extraction and representation, CNN design and training, and decision fusion. In the preprocessing step, each word in the tweets was processed by creating tokens and removing alphanumeric characters. Stop words were removed using a rule-based method and inference was used to obtain the root word of each token. Three types of embedding vectors were used for each neat token: fast text-based embedding, probability-based domain-specific embedding, and probability-based domain-specific embedding. The fastText embedding vector (ft) was pre-trained with multilingual datasets, domain-agnostic embedding (da) used a dataset from the opposite domain (NepaliNewsDataset), and domain-specific embedding (ds) relied on specific domains. Python tools Sklearn and Keras were used to implement the proposed methods. The researchers designed ten different train/test sets, with each category split 70:30, and reported peak performance ratings from over ten runs for analysis. The proposed ensemble method achieved an accuracy of 68.7% in their experiments.



**Table 1: Summary of Existing ML Sentiment Analysis models**

Ref #	Year	Author	Dataset	Classes	Epochs	Features & Classifiers	Accuracy
[1]	2020	Sharma, A. and Daniels	Twitter API Dataset Indian elections	Positive/Negative	-	Random Forest	86.87%
[2]	2017	Rezaeinia et al.	Covid-19 Twitter API Dataset	Positive/Negative	-	Improved Word Vector	-
[3]	2018	B. Oscar Deho et al.	Twitter API Dataset on US base	Positive/Negative	-	Random Forest classifier	81%
[4]	2017	A. N. Farhan and M. L. Khodra	TripAdvisor Dataset on hotels and restaurants	Positive/Negative/Neutral	-	SVM Word2Vec C&W	83% 75% 73%
[5]	2019	H. Imaduddin et al.	Traveloka dataset of hotels and restaurants reviews	Positive/Negative	-	Glove model- Word2vec CBOW - Word2vec skip-gram Doc2vec-.	95.52% 92.72% 91.81% 94.81%
[6]	2017	M. Al-Amin et al.	Bengali Twitter API Dataset	Positive/Negative	-	word2vec and words' sentiment extraction	75.5%
[7]	2020	W. Yue and L. Li	Quora's Internet data set	Positive/Negative/Neutral	-	Word2vec-CNN-BiLSTM	91.48%.
[8]	2016	Q. Li et al	Twitter API Dataset	Positive/Negative/Neutral	-	LibLinear and SMO	66.8%
[9]	2017	Y. Sharma		positive / negative	-	GloVe	-
[10]	2016	Ren et al.	Twitter API Dataset	Positive/negative	-	Latent Dirichlet Allocation	81.02%
[11]	2021	Sitaula, C	Nepali's Twitter sentiment database	positive, neutral, and negative	-		68.7%

### 3.1.2 BERT

Rifat et al. [24] implemented a novel deep learning model called Bidirectional Encoder Representations (BERT) from the Transformers model to analyze Covid-19 Twitter tweets. The authors began by performing basic pre-processing steps, such as removing irrelevant symbols, URLs, and mentions from the tweets. They then utilized the BERT model to extract features from the data and trained a classification model to classify the tweets into five sentiment classes: positive, extremely positive, negative, extremely negative, and neutral. The model's performance was evaluated using a dataset comprising 3798 tweets, and the BERT model achieved an accuracy of 87.57% in sentiment classification. The researchers also compared the results of the BERT model with six other state-of-the-art models, namely Logistic Regression, Support Vector Machine, Stochastic Gradient Descent, XGBoost, Random Forests, and Naive Bayes. The comparative analysis demonstrated that BERT outperformed all other models, with the models' achieving accuracies of 60%, 56%, 56%, 54%, 50%, and 44% respectively.

Mahor and Manjhar [25] evaluated the public's sentiments during the Covid pandemic using BERT model. The research analysis was done on Twitter dataset containing 16,000 covid-19 related tweets from February to May 2020 and then manually labelled them as positive, negative, and neutral. The raw tweets dataset underwent preprocessing via reducing noise (removing URLs, special characters, hashtags stop words), tokenization, and normalization. BERT model was then used to automatically classify the sentiment of the tweets. The model's performance was assessed using different evaluation metrics, such as accuracy, precision, recall, and F1-score. The evaluation revealed that the BERT model exhibited strong performance, achieving an accuracy of 82.1% in categorizing tweets. The model demonstrated effectiveness in classifying negative tweets specifically related to Covid-19. The tweets often discussed common topics such as government policies, personal protective equipment, and social distancing were also analyzed. The study presented the NLP and ML techniques' potential in sentiment analysis towards health emergencies and findings can be effective in public health communication and policymaking decisions.

Kumar et al. [26] explored the importance of sentiment analysis in various fields, particularly on social media platforms. To address this issue, the authors introduced a novel approach that utilizes machine learning (ML) and the BERT model for Twitter data sentiment analysis. Proposed methodology consisted of two main phases: pre-processing and classification. During the pre-processing phase, the Twitter dataset obtained from Kaggle was subjected to

cleaning and normalization techniques such as stop word removal, stemming, and emoji handling. In the classification phase, features were extracted from the pre-processed tweets using the BERT model, and the ML Logistic Regression algorithm was employed to classify the tweets into three sentiment categories: positive, negative, or neutral. The performance of the proposed BERT model was compared to several existing state-of-the-art models, including Random Forest, XG Boost, Logistic Regression, SVM, Stochastic Gradient Descent - Classifier (SGD-Classifier), and Decision Tree. Evaluation metrics such as accuracy, precision, recall, and F1-score were employed to assess the models. Among the ML methods, Logistic Regression achieved the highest accuracy, with a score of 81.74%. However, the BERT model showed exceptional results in comparison with all ML models with an accuracy of 93%. The authors proposed that the presented methodology can help in multiple applications like monitoring the brand, analyzing customers' feedback or social media content.

In the research [27], Topbas et al. presented the sentiment analysis of the public regarding the Covid pandemic on Twitter platform using two deep learning models i.e., Recurrent Neural Network (RNN) and BERT. The authors showed through the results that both RNN and BERT deep learning models are effective in the sentiment analysis of the text. The dataset contained the tweets relevant to Covid-19 pandemic which were collected through the Twitter API and then on their sentiments' basis, the collected tweets were manually labeled as negative, positive, or neutral. For the proposed methodology, the Twitter dataset was then pre-processed using the lowercasing, tokenization, removal of URLs and stop words functions. The first model, RNN, the variant of LSTM model was used. The model was trained on the processed data and RNN model achieved an accuracy of 77.5% on the test dataset. The second model transformer-based Bert model was used. The pre-trained BERT model was fine-tuned on the processed dataset and model achieved 86.4% accuracy on the 10% test dataset, which outperformed the RNN model. The BERT results showed that it also classified the tweets with mixed sentiments which shows that model has distinct capability to capture text sentiment. The feature importance analysis done in the study showed that the "lockdown" and "death" words had influence on the tweets' sentiment. The paper results also provided insights into the significant words that affected sentiments in covid pandemic tweets.

Nair et al. [28] conducted a study focusing on analyzing Twitter data to gain insights into public sentiment surrounding the Covid-19 pandemic. The research aimed to compare

sentiments across different time periods and locations. The authors collected approximately two million Covid-19-related tweets from March to May 2020. Natural Language Processing (NLP) techniques were employed to classify the tweets into positive, negative, or neutral sentiments. The study utilized three different algorithms, namely Logistic Regression, BERT, and Vader, for sentiment analysis of the Covid-19 tweets. The results indicated that the BERT model outperformed the other models, achieving a higher accuracy rate of 92%. The study results showed that the overall public sentiment in the Twitter tweets towards the pandemic was negative as the negative tweets were in higher number as compared to the other positive or neutral tweets. The sentiment analysis showed that public sentiment was more negative towards pandemic in the March and April months than in month of May. Moreover, among the geographical regions, more negative sentiment was found in countries where pandemic reported cases were higher and fatal. The study implied that Twitter is a helpful tool in getting insights about public sentiments for any current hot topic or any pandemic. Also, the sentiment depends on multiple factors like geolocation and time zone.

Chintalapudi et al. [29] conducted a study on sentiment analysis of Indian tweets related to Covid-19, employing deep learning models. The authors utilized CNN, LSTM, and BERT models to classify Indian tweets from Twitter. A dataset consisting of approximately 1.5 million Indian tweets related to the Covid-19 pandemic was collected for the research. The collected data underwent preprocessing and manual labeling into four classes: fear, sadness, joy, and anger, based on the content of the tweets. The dataset was then divided into three sets: train, validate, and test, which were utilized for training and evaluating the CNN, LSTM, and BERT models. The performance of the models was evaluated using metrics such as accuracy, precision, recall, and F1-score. The findings demonstrated that the fine-tuned BERT model outperformed the other models, achieving the highest accuracy of 89%. The analysis of the results concluded that the negative tweets were more dominant in the dataset, followed by neutral tweets and then the positive tweets. The analysis showed that deep learning models are more effective in sentiment classification and highlighted the significance of analyzing data on social media during covid19 pandemic.

**Table 2: Summary of Existing Sentiment Analysis models using BERT.**

Ref #	Year	Author	Dataset	Classes	Epochs	Features & Classifiers	Accuracy
[24]	2022	Rifat et al.	Kaggle Corona NLP Dataset	Pos/Neg/Neu/ Extremely Neg/ Extremely Pos	65	BERT LR Model	88% 60%
[25]	2022	Mahor et al.	Covid-19 Twitter API Dataset	Positive/ Negative/ Neutral	-	BERT RF Model	70% 59
[26]	2021	Kumar et al.	Kaggle Sentiment140 Dataset	Positive/ Negative	20	LR Model BERT	81.74% 93.63%
[27]	2021	Topbas et al.	Covid-19 Twitter API Dataset	Positive/ Negative/ Neutral	10	RNN BERT	86.4% 83.14%
[28]	2021	Nair et al.	Covid-19	Positive/ Negative/ Neutral	-	BERT LR Model	92% 83%
[29]	2021	Chintalapudi et al.	GitHub CoViD-19- tweets	Fear/ Sad/ Anger/ Joy	-	BERT LR Model	89% 75%

### 3.2 Research Gaps

The current analyzing tweet polarity methods mostly are lexicon-based or machine-based, but these methods don't conserve the contextual semantics of words in the text. Many existing learning-based approaches focused on producing functional features while some researchers used word-based features, such as TF-IDF. Also, the prior work indicated that deep learning approach has been rarely used in sentimental analysis of Covid Twitter Data. Mainly the developed models predicted accuracy among the three main sentiment classes.

### **3.3 Research Contributions**

The main contributions of the underlying research are:

- A novel Twitter data set related to Covid-19 is collected from Twitter platform.
- The sentiment of the tweets is captured through deep word embedding along with MiniLM by conserving the semantics of the words.
- A detailed sentiment analysis is proposed for five classes i.e., Positive, Extremely Positive, Negative, Extremely Negative and Neutral.
- A detailed ablation study is conducted to experiment with different available models.
- Analyzing public's reactions through Story Generation and Visualization from Covid Tweets.

## CHAPTER 4: METHODOLOGY

### 4.1 Data Pre-Processing

In data mining, the most crucial phase is the preprocessing of the data which involves the data transformation and data preparation for extracting the required knowledge. Data preprocessing step involves different methods like data cleaning, transformation, integration, and dataset reduction. As a result of data preprocessing, cleaned or structured dataset is produced which helps in modeling. The data in its raw form, obtained from several resources, is not reliable for analysis. So, it's necessary to clean the raw data before the analysis stage. In analytic projects, almost 70% of the project's work involves data cleaning. Data preprocessing is a tiresome but unavoidable task.

The tweet data from date 16<sup>th</sup> March 2020 to 14<sup>th</sup> April 2020 has been extracted for current research. The extraction of the tweet data was restricted to three fields i-e; tweet id, the tweet date, and the tweet text which is to be analyzed. Data preprocessing was done first to clean the raw tweets for the acceptance of model. The data contains two columns i.e., tweet id and text. In the preprocessing step, the text column is only preprocessed. For this, a cleaner function is created with all the expressions that removes the links, hashtags, user mentions, special characters, spaces etc. All the processing steps are done and reflected in the Original Tweet column. The preprocessing is done on the text column in the data. The following preprocessing steps are done for cleaning the tweets raw data.

#### 4.1.1 Removing Twitter Links

In the first step of data preprocessing, URL links are removed from the Twitter tweets as they contain irrelevant characters which will not contribute in our aim to classify the tweets according to their sentiments. The pattern "`(\w+:\w+\S+)`" is used to remove the links from tweets, in the cleaner function.

#### 4.1.2 Removing Twitter Hashtags

After removing the links from tweets, hashtags are then removed from the tweets in preparing the tweet data for classification. To remove the hashtags, "`(#[A-Za-z0-9_]+)`" is used as pattern to the cleaner function. The pattern specifies all the words starting with '#' and removes them from the tweet.

### **4.1.3 Removing Twitter Handles (@user)**

In the next step of data preprocessing, twitter handles are removed as because of the privacy concerns they are already disguised as @user which aren't useful for analysis. These twitter handles barely provide any information about the tweet's nature. To remove the twitter handles, “(@[A-Za-z0-9\_]+)” is used as pattern to the cleaner function. It's a regular expression that selects all the words starting with '@' and removes them from the expression.

### **4.1.4 Removing Punctuations, Numbers, and Special Characters**

During this preprocessing step, punctuation marks, numbers, and special characters are eliminated as they do not contribute to differentiating the various types of tweets. To accomplish this, punctuation, numbers, and special characters within the tweets are replaced with spaces. The regular expression "[^a-zA-Z#]" is employed, which encompasses all characters excluding alphabets and hashtags (#).

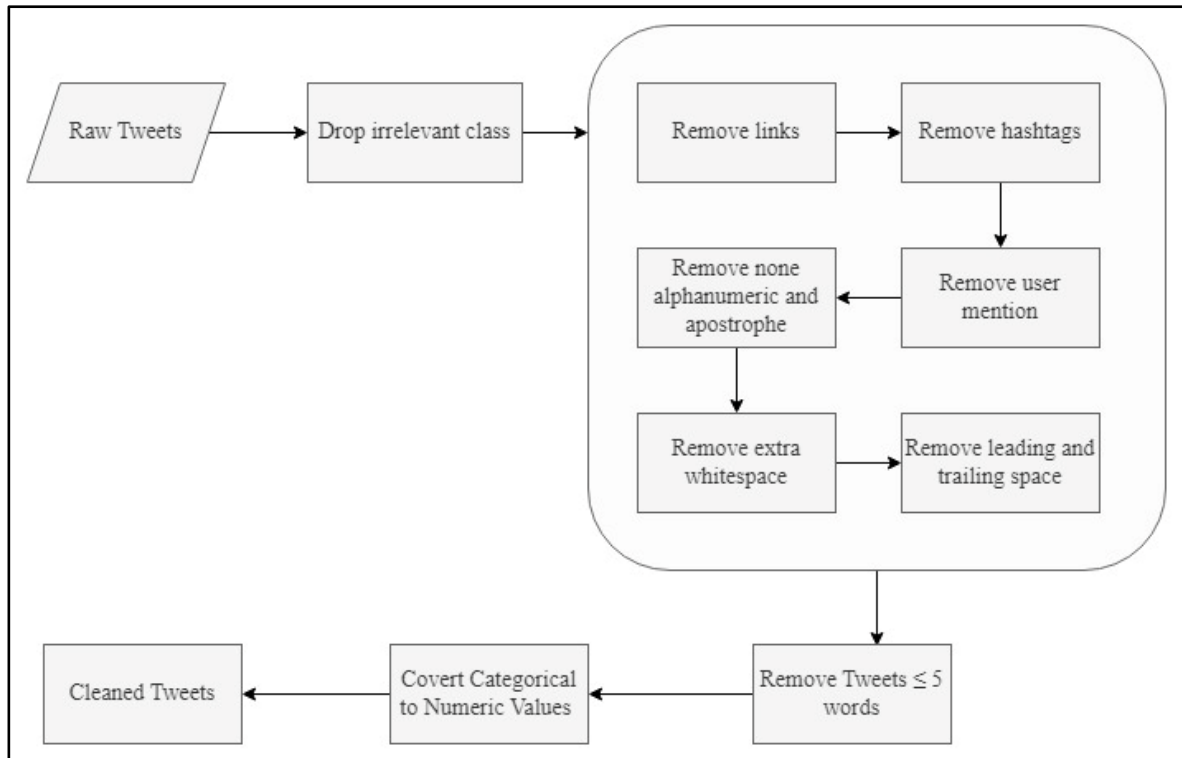
### **4.1.5 Removing White Spaces**

The next step in pre-processing included the removal of white spaces. The quantity of data that necessitates processing for the particular undertaking is reduced by removing the redundant white spaces. There is a character limit in tweets so it's important to make most of the available spaces by removing extra white spaces from tweets to keep them within the character limit. It also makes the text easier in analyzing and in tokenizing text into distinct words which is important in sentiment analysis and text classification tasks.

### **4.1.6 Removing Short Tweets**

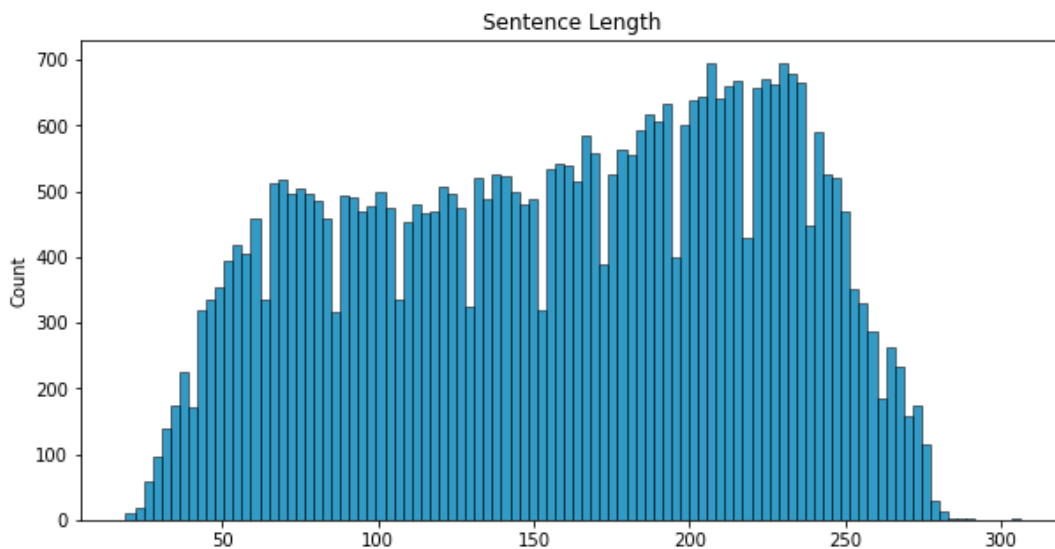
The short text generally does not add any considerable value to the results. For example, tweets like “Is this effective?”, “Is it possible?” are called short tweets. As they won't support classifying the tweets' nature so, such tweets are removed from the tweet's dataset. In the process of removing the short tweets, the crucial parameter lies in determining the length of the textual words that are to be eliminated. In this research, all tweets that have length 5 or less are removed from the dataset. Five words tweets are the least significant, so it's better to discard them.





**Figure 3: Overview of Steps involved in Preprocessing.**

After preprocessing steps, a significant difference between the raw tweets and processed tweets is observed in Table 3. Only the important or significant words are preserved in the tweets and the noise (twitter handles, numbers, punctuations, special characters, and short words tweets) have been discarded from the raw tweets. The figure 4 graph shows the maximum sentence tokens in the data corpus i.e., 306.



**Figure 4: Maximum Sentence Tokens in Dataset**

**Table 3: Comparison of Raw and Processed Tweets**

OriginalTweet - Raw	OriginalTweet - Processed
advice Talk to your neighbours family to exchange phone numbers create contact list with phone numbers of neighbours schools employer chemist GP set up online shopping accounts if poss adequate supplies of regular meds but not over order	advice Talk to your neighbours family to exchange phone numbers create contact list with phone numbers of neighbours schools employer chemist GP set up online shopping accounts if poss adequate supplies of regular meds but not over order
Coronavirus Australia: Woolworths to give elderly, disabled dedicated shopping hours amid COVID-19 outbreak <a href="https://t.co/bInCA9Vp8P">https://t.co/bInCA9Vp8P</a>	Coronavirus Australia Woolworths to give elderly disabled dedicated shopping hours amid COVID 19 outbreak
My food stock is not the only one which is empty... PLEASE, don't panic, THERE WILL BE ENOUGH FOOD FOR EVERYONE if you do not take more than you need. Stay calm, stay safe. #COVID19france #COVID_19 #COVID19 #coronavirus #confinement #Confinementtotal #ConfinementGeneral <a href="https://t.co/zrlG0Z520j">https://t.co/zrlG0Z520j</a>	My food stock is not the only one which is empty PLEASE don't panic THERE WILL BE ENOUGH FOOD FOR EVERYONE if you do not take more than you need Stay calm stay safe
Me, ready to go at supermarket during the #COVID19 outbreak. Not because I'm paranoid, but because my food stock is literally empty. The #coronavirus is a serious thing, but please, don't panic. It causes shortage... #CoronavirusFrance #restezchezvous #StayAtHome #confinement <a href="https://t.co/usmuaLq72n">https://t.co/usmuaLq72n</a>	Me ready to go at supermarket during the outbreak Not because I'm paranoid but because my food stock is literally empty The is a serious thing but please don't panic It causes shortage
As news of the region's first confirmed COVID-19 case came out of Sullivan County last week, people flocked to area stores to purchase cleaning supplies, hand sanitizer, food, toilet paper and other goods, @Tim_Dodson reports <a href="https://t.co/cfXch7a2IU">https://t.co/cfXch7a2IU</a>	As news of the region s first confirmed COVID 19 case came out of Sullivan County last week people flocked to area stores to purchase cleaning supplies hand sanitizer food toilet paper and other goods reports

## 4.2 Feature Selection

### 4.2.1 Word Embedding

In the field of Natural Language Processing (NLP), word embeddings have emerged as an exciting and prominent trend. They are widely used to represent words as vectors. The primary objective of word embeddings involves conversion of high-dimensional word features into lower-dimensional feature vectors while maintaining the contextual similarity of

the corpus. Word embeddings provide a learned textual representation, where words with similar meanings have similar vector representations. In word embeddings, each unique word is mapped to a real-valued vector in a pre-defined vector space. This mapping is achieved through a learning process, often resembling a neural network, where the values of the vectors are adjusted. Typically, word vectors have tens or hundreds of dimensions, which is in contrast to sparse word representations like one-hot encoding, which can have thousands or millions of dimensions.

Word embeddings offer a convenient approach to working with textual data. They extract meaningful features from the text, which can then be used as input for machine learning models. Unlike other feature extraction methods such as Bag of Words (BOW), TF-IDF, or CountVectorizer, word embeddings preserve both semantic and syntactic information of the text. These methods rely solely on word frequency in the text, and their vector size for each word is equal to the number of words in the dataset or vocabulary. If the majority of the elements are zero in vector, then it will result in a sparse matrix. If the size of input vectors is large, then many weights are produced and as a result high results are needed for training the model. All these problems are solved by word embeddings.

The three main advantages of using word embeddings compared to other feature extraction methods like Bag of words or TF-IDF are:

1. Reduction in dimensionality as it reduces a significant number of features that are needed to build the model.
2. It captures the word meanings, semantic relationships, and captures the different contextual forms in which words are used in the sentences.
3. It also helps in predicting words around a specific word.

The two most well-known methods used for word embedding are Word2Vec and GloVe. The research showed these both methods are effective approaches for learning vector representation of the words. For this, NLP tasks like word calculating, words similarity uses both methods. But it's difficult to choose between these two methods.

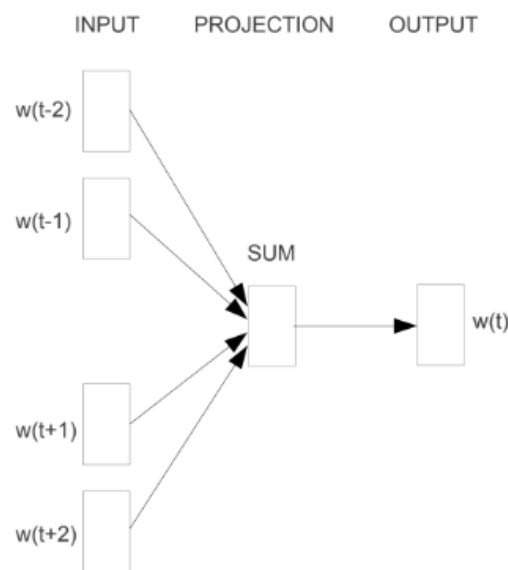
#### **4.2.2 Word2Vec**

Word2Vec is a widely recognized word embedding method developed by Google. It encompasses a set of models used to generate word embeddings, which are vector representations of words that prove useful in various natural language processing tasks. The Word2Vec model consists of two neural network layers that process textual content. It takes

textual documents such as articles or tweets as input and produces a set of vectors with multiple dimensions as output. Mikolov et al. [23] introduced the Word2Vec model for word embedding and described two different training modes: CBOW (Continuous Bag-of-Words Model) and Skip-gram (Continuous Skip-gram Model). These training techniques involve narrow neural networks that map words to a target variable, which can be another word or a set of words. Through these methods, words are represented as vectors by learning the weights associated with each word.

### 1. CBOW

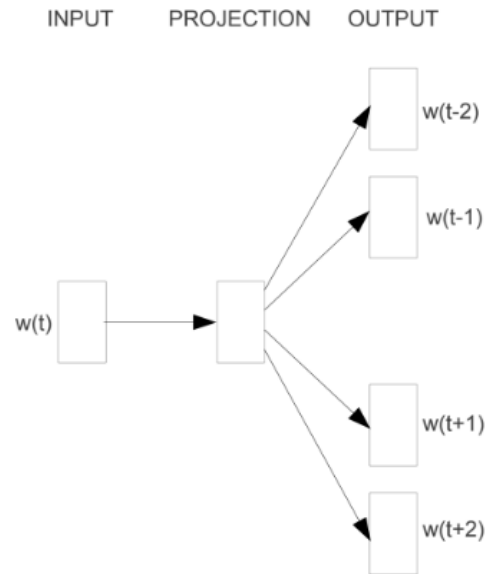
The model predicts a target word from the given context or words around the target word by predicting its probability in the context in which it is being used in. In simple words, it can be said that CBOW approach is equivalent to fill in the blank method i.e., take out a word from the sentence and then its asked to guess the word (target). Figure 5 displays the CBOW model architecture.



**Figure 5: CBOW Model Architecture**

### 2. Skip-gram

The skip-gram method works in the reverse of the CBOW method. The aim of the model is to predict the context window of words using the given word. In simple words, its equivalent to a word is given and then its asked to guess the words before and after to the given term. Figure 6 displays the Skip-gram model architecture.



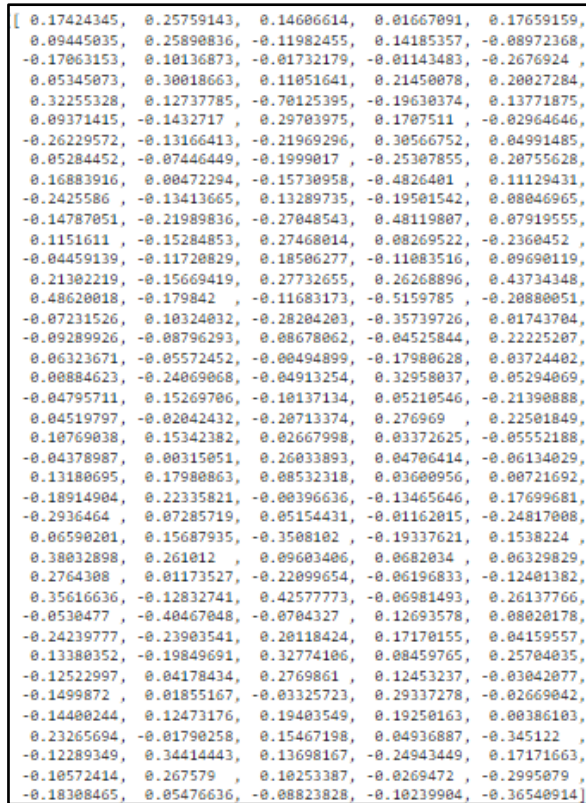
**Figure 6: Skip-gram Model Architecture**

Both the CBOW and Skip-gram models in Word2Vec utilize artificial neural networks. Initially, each word in the vocabulary is assigned a random  $n$ -dimensional vector. During the training process, the neural network algorithm learns the optimal vector representation for each word using either the CBOW or Skip-gram approach. Figure 3 and Figure 4 provide architectural illustrations of the word2vec CBOW and Skip-gram models, respectively.

In our approach, we have chosen Skip-gram model due to its specific advantages over the CBOW method.

- Skip-gram model creates two vector representations of each word where required by capturing two meanings/semantics for each word. For example, there will be two vectors created for the word “Apple”, one for fruit representation and one for technology representation.
- Skip-gram with negative sub-sampling surpasses the CBOW method in performance.

The significant step is the training of word2vec model, to get word vector  $n$ -dimensional representation for each distinctive word present in the data. For this, there are two approaches i.e., one is to use pre-trained word vector representations and the other is to create our own word vector representations by training the model. In this research, we have trained our own vectors using the model and skip-gram algorithm as the pre-trained word vector size is generally very large. The visual representation in Figure 7 displays the vector 200-dimensional representation of the word "food" from the vocabulary.



**Figure 7: 200-Dimensional Word Vector Representation for Word “Food”**

Training word2vec model showed that it finds the best possible similar words for the target word as the model has captured the semantic meaning for all the words and finds the best similar vectors by using cosine similarity. Table 4 shows the most similar words returned by word2vec model from the corpus for the target word “dinner”.

**Table 4: Most similar words for the target word “Dinner” Generated by Word2Vec**

Similar Words	Similarity/Probability
brisket	0.565
flatbread	0.521
slug	0.520
ragu	0.520
tasti	0.505
gnocchi	0.502
picki	0.498
lightbulb	0.495
taco	0.490
haircut	0.488

Now, once the word vector representations are created through word2vec, the next step included to create vectors for tweets as our dataset contained tweets instead of words. Just by taking the mean of all the word vectors present in the target tweet, we created a vector representation for a complete tweet. The resultant tweet vector length is the same as the individual word vector length i.e., 200. This process is repeated for all the tweets present in the data to get the tweet vector representation for the entire tweet dataset.

## **4.3 Algorithm**

### **4.3.1 Machine Learning (ML) Model**

There are multiple Machine Learning algorithms that can be used for classification, but the following algorithm was selected for this research.

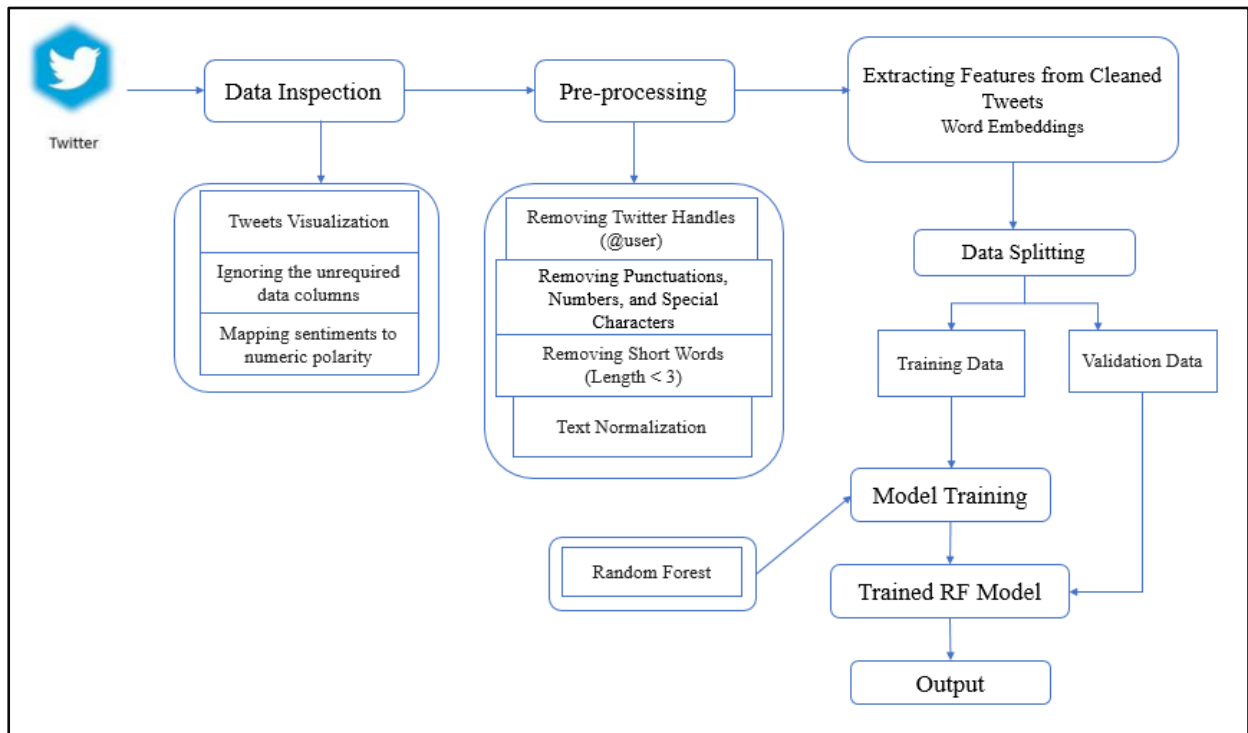
#### **4.3.1.1 Random Forest**

Random forest algorithm is a supervised ensemble machine learning method that is used for various tasks like feature selection, regression and classification tasks. The method involves combining several weak learner models to form a stronger classification prediction model. It achieves this by aggregating multiple decision trees to create a random forest, in which the individual trees act as the weak learners. Each decision tree model is trained on the concept of random sampling of the dataset and random subset of features. The random sampling basically helps in reducing the model to overfit and enhances the model's performance.

The working [1] of the Random Forest model can be summarized as follows:

1. A subset of N samples is randomly selected from the given dataset with replacement as training set.
2. A subset of features is randomly selected from the existing features set.
3. Decision tree is then built using selected samples subset and feature subset.
4. A large number of multiple decision trees are built by repeating steps 1-3.
5. Each tree predicts the class label/output independently for the input data.
6. The final prediction is determined by combining the results of all decision trees. For classification tasks, the majority voting method is used, where the class with the highest number of votes is selected as the final prediction. For regression tasks, the average prediction method is used, where the average of all individual tree predictions is taken as the final prediction.

The sentiment classification in this study utilized the Random Forest model, which is known for its ability to mitigate overfitting and provide improved outcomes by combining the results from multiple decision trees. The model was trained using 400 decision trees and the word vectors generated from the word2vec model. Once trained, the model was employed to predict the sentiment polarities of the tweets in the test dataset. Figure 8 presents the flow chart illustrating the proposed Random Forest model for Twitter sentiment classification.



**Figure 8: Proposed RF Model for Twitter Sentiment Classification**

## 4.3.2 Deep Learning (DL) Models

### 4.3.2.1 Bidirectional Encoder Representations from Transformers

BERT (Bidirectional Encoder Representations from Transformers) is a deep learning model that was introduced by Google in 2018 [30]. It is a pre-trained language model based on the transformer architecture, which is specifically designed for sequential data processing, such as text. BERT has undergone pre-training on a large dataset of unlabeled text, allowing it to learn the contextual relationships and meanings of words. One notable aspect of BERT is its bidirectionality. Unlike other unidirectional language models, BERT considers both preceding and succeeding words when predicting the next word in a sentence. This enables BERT to capture a broader context and understand the dependencies between words. Additionally, BERT takes into account the position and order of words within a given



context, distinguishing it from models like Word2Vec. For instance, apparently in Word2Vec there isn't any difference in the two sentences "Alvin likes Amy" and "Amy likes Alvin", but they do have difference in BERT. BERT is a pre-trained model that estimates the frequency of words in the sentences and skillfully predicts the next sentence on the basis of the preceding sentence. The BERT model is used to classify the sentiments by considering the context of a word in which it is used in the sentence. This helps in determining the correct meaning of the word.

BERT has been proved as a highly effective model for a wide range of natural language processing tasks, such as sentiment analysis, text classification, and question-answering [30]. It is also very flexible and can be fine-tuned to a smaller labeled data set to produce conventional model to fit a specific NLP task. BERT can be fine-tuned for a specific task by adding one extra layer at the output. Overall, BERT is a powerful natural language processing tool that has greatly improved the accuracy and efficiency of many language applications.

#### **4.3.2.1.1 Model Overview**

The BERT model's architecture is built on Transformers. Multi-layer bidirectional transformer encoders are used in BERT model to represent languages. There are two types of BERT models based on the model's depth i.e., BERTBase and BERTLarge. The BERTBase model has 12 transformer block layers with 768 hidden size, 12 self-attention heads, and approximately 110M trainable parameters. In contrast, the BERTLarge model has 24 transformer block layers with 1024 hidden size, 16 self-attention heads, and approximately 340M trainable parameters. Regardless of the task, whether it is NLI, classification, or Question-Answering, BERT uses the same model architecture with minimum number of modifications i.e., an extra output layer is added for classification for the respective task.

#### **4.3.2.1.2 Input Output Format**

The importance of dataset preprocessing lies in transforming the input raw data in such a format that BERT can easily understand and process that input data. For the BERT model, preprocessing steps are divided into three levels i.e., Tokenization, Segmentation, and Word Ordering [32].

## **I. Token Embedding**

The entire input to BERT must have a single sequence. The BERT model utilizes special identification tokens, namely [CLS] and [SEP], to effectively comprehend the input it receives. The [CLS] token serves as a special classification token, and the final hidden state of BERT associated with this token, is employed for classification purposes. On the other hand, the [SEP] token serves as a separator token and must be included at the end of one input. The [SEP] token assists the BERT model to recognize the termination of first input and the beginning of the next sentence in the same input sequence. Tasks like NLI and QA tasks require more than one input, so [SEP] token marks the separation among the input sentences accordingly. Wordpiece embeddings are used by BERT for input tokens.

## **II. Segment Embedding**

BERT uses segmentation embedding to differentiate between the multiple input sentences. The embedder component in the BERT model takes the tokens obtained from the previous step and distinguishes whether those tokens belong to the first input sentence or the second input sentence. Tokens of sentence 1 will have predefined embeddings of 0 and tokens of sentence 2 will have segment embeddings as 1 (figure 9).

## **III. Positional Embedding**

To capture the positional information of tokens in the input sentences, BERT utilizes positional embedding. This enables the model to understand the relative positions of the tokens within the sequence. The embedder component in BERT generates positional embeddings that indicate the position of each token in the input sequence, which can then be input to BERT for model's pretraining.

Consider a tweet with two sentences, "Do Not Panic. Do Not Fear." (figure). Tokenization would change the tweet input as ([cls], Do, Not, Panic, [sep], Do, Not, Fear). The model takes input final embeddings which are the sum of all three embeddings i.e, token embedding, segment embedding and position embeddings. These final embeddings are fed into the model to get the output after training.

Input	[CLS]	Do	Not	Panic	[SEP]	Do	Not	Fear	[SEP]
Token Embeddings	$E_{[CLS]}$	$E_{Do}$	$E_{Not}$	$E_{Panic}$	$E_{[SEP]}$	$E_{Do}$	$E_{Not}$	$E_{Fear}$	$E_{[SEP]}$
	+	+	+	+	+	+	+	+	+
Segment Embeddings	$E_0$	$E_0$	$E_0$	$E_0$	$E_0$	$E_1$	$E_1$	$E_1$	$E_1$
	+	+	+	+	+	+	+	+	+
Position Embeddings	$E_0$	$E_1$	$E_2$	$E_3$	$E_4$	$E_5$	$E_6$	$E_7$	$E_8$

**Figure 9: BERT Tokenization and Vectorization**

#### 4.3.2.1.3 BERT Pre-Training

BERT has been already trained on the two unsupervised datasets such as Wikipedia and Book corpus using language modeling techniques. The pre-training of the BERT’s model [31] consisted of two tasks namely Masked Language Model (MLM) and Next Sentence Prediction (NSP).

##### I. Masked Language Model

In BERT’s pre-training, one half of the process is Masked Language Model training that basically optimizes the weights inside BERT to get the original sentence [32]. Firstly, the whole corpus is broken down into tokens, to which meanings can be assigned. Then, identical vectors are created by copying the tokens which helps in calculating the model’s loss and optimization. Then, 15% of the randomly selected tokens in the corpus sequence are masked, so the model can be trained to correctly predict those masked words via the words’ context surrounding them. For instance, the tweet sentence “Do not panic” would be symbolized as “Do not [MASK]”. Then the BERT model will be trained to predict the [MASK] word as “panic”. In general, all the [MASK] words in the corpus aren’t replaced with [MASK] word as this token doesn’t always appear in fine-tuning. For this, 80% of the words in corpus as masked as [MASK] token, 10% are masked with random words and the rest 10% words are kept as the same original words. Lastly, the model’s loss is calculated by measuring the discrepancy between the predicted probability of every masked token and its true value. MLM process iterates till the convergence of the loss function.

## II. Next Sentence Prediction

Next Sentence Prediction is the other half of the BERT's pre-training process which is used to capture the correlation between pairs of sentences in the input. NSP refers to a binary classifier which takes two sentences as input into the BERT model and predicts whether they have a meaningful or sequential relation or if their relationship is just random [32]. In NSP task training, 50% of the corpus data is marked as "isNextSentence", which means the second sentence (sentence B) in the input sequence is simply the next sentence of the first sentence (sentence A) in the dataset. Whereas the other 50% of the sentences in dataset is marked as "NotNextSentence" which means sentence B isn't the succeeding sentence of sentence A but is any random sentence in the corpus. The correct label is predicted using the hidden output state related to the [CLS] token and the loss is calculated. After the pre-training phase, the BERT model can be further fine-tuned on task-specific datasets.

During the BERT training, the model optimizes the combined loss function of MLM and NSP by training both tasks simultaneously on the easily available English Wikipedia and Toronto Book Corpus datasets.

### 4.3.2.1.4 BERT Fine Tuning

Fine tuning the BERT model can produce desired results for the respective task. Following steps were followed using the Hugging Face [33] along with PyTorch library for the sentiment classification using the pre-trained BERT base model (figure 10).

- A train-test split method was used to split the dataset into training and validation sets. The dataset was partitioned such that 85% of the data was allocated for model training, while the remaining 15% was used to evaluate model performance. To address any potential class imbalance within the dataset, the stratify parameter was used during the partitioning process. This ensured that the proportion of samples in each class was preserved in both the training and validation sets. Table 5 shows train-validate split for each class in the dataset.
- BERT tokenizer is employed to tokenize the text data before feeding it into the model.
- The use of deep neural networks can significantly benefit the sentiment analysis tasks, via achieving notable results by converting the input training data into torch tensors. This is achieved via embedding technique which transforms the input sentences into embedded vectors. Transformer encoders are used. The model is then

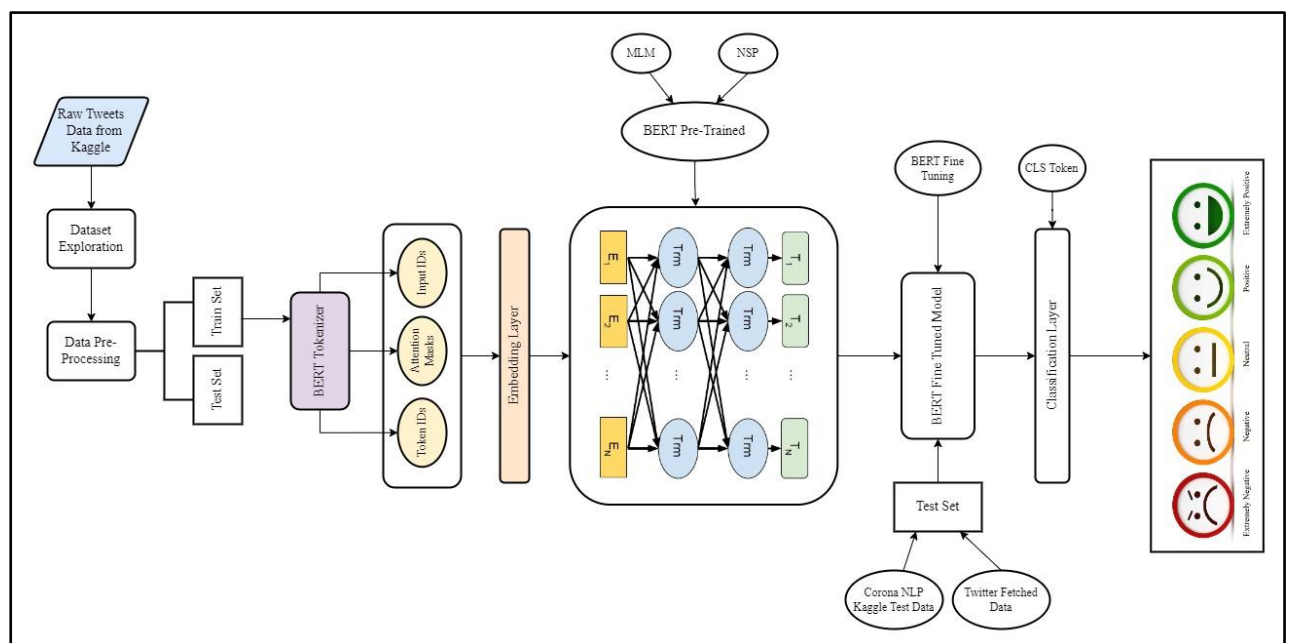
created, and each sentence passes through it after being encoded as a vector of 512 elements.

**Table 5: Splitting of Train-Val Sets for Each Sentiment Class**

Sentiment	label	data_type	Username	Screen Name	Location	Tweet At	Original Tweet
Extremely Negative	1	train	4636	4636	3597	4636	4636
		val	824	824	635	824	824
Extremely Positive	4	train	5615	5615	4443	5615	5615
		val	993	993	819	993	993
Negative	3	train	8329	8329	6520	8329	8329
		val	1464	1464	1151	1464	1464
Neutral	2	train	6127	6127	4902	6127	6127
		val	1081	1081	880	1081	1081
Positive	0	train	9558	9558	7625	9558	9558
		val	1685	1685	1333	1685	1685

- The text data, which consists of tweets in this specific case, is treated as a sequence and processed through a pre-trained BERT model using the "BertForSequenceClassification" class from the Transformer library. The chosen pre-trained model for this task is "bert-base-uncased," and it is configured to train on the provided dataset.
- The data must be batched in order to train the data. The process of batching can be automated using the data loaders. In the current model, RandomSampler is used for both batch training and validation dataset. The Random Sampling method offers variation in the data by ensuring that each batch of training data appears in a random order. Multiple experiments have been conducted with different batch sizes and concluded that the batch size of 4 produced better results for the current dataset.

- In deep learning approaches, several attributes, for example weights and learning rate can be adjusted in order to reduce the training and validation loss. These attribute values can be modified through an optimizer. In this study, AdamW optimizer has been used due to its capability to adapt step sizes for individual weights. A few experiments have been conducted in this study to find the best possible learning rate following the BERT official documentation [30] and found  $1e-5$  learning rate produced better results with our current dataset. A scheduler is employed to manage processes over a specific time frame. Our scheduler included a learning rate that decreases linearly after a linear inflation during a warmup period. To prevent overfitting, we determined that the model should not exceed 10 epochs.
- Model performance is assessed using the accuracy metric.



**Figure 10: Flow Chart of Proposed BERT Model**

#### 4.3.2.2 Distil BERT

DistilBERT is a pre-trained language model that utilizes the transformer architecture, similar to BERT. Distil BERT is smaller in size [34], faster, a smaller vocabulary size and lighter than BERT, with half the number of layers. This design with less memory requirements and simple model structure makes the Distil BERT model more efficient and trivial compared to BERT, particularly in situations with limited computing resources. Despite its smaller size, it still retains 97% of the language processing power of BERT and can perform several natural language processing tasks. Compared to BERT, Distil BERT is 40% smaller and 60% faster,

making it a popular choice for applications where fast training and conclusion times are important, such as mobile or embedded devices and low-resource environments. While the learning rate can have an impact on the performance of the model, the difference in accuracy of the model between different learning rates is not significant.

The efficiency of Distil BERT model is attained by using knowledge distillation process, which focuses on replicating the larger BERT model's behavior during the process of model training. This process allows the model to inherit the knowledge acquired by the larger BERT model and achieve a comparable level of performance. Despite its smaller size, various studies [24] showed that the Distil BERT model also shows almost similar performance to larger BERT model on several NLP tasks.

#### **4.3.2.3 DeBERTa-v3-base**

The DeBERTa v3 is a newly developed generative AI model in the field of natural language processing. This model belongs to the DeBERTa family of models and has been considered as a benchmark for various NLP tasks due to its exceptional performance. The DeBERTa-v3-base model from Microsoft [35] is a transformer-based language model which is an improved version of the original DeBERTa model and includes improvements such as disentangled attention mechanisms and dynamic token masking. The Microsoft DeBERTa-v3-base model consists of 12 layers and 768 hidden units with a vocabulary size of 128K words, which makes it smaller and faster than its original larger DeBERTa-v3 model. This trivial model still shows high performance on several natural language processing tasks.

The architecture of this model is built on the transformer-based language model that uses self-attention mechanisms for processing sequential data. However, it includes several enhancements to the original model architecture, such as improved positional embeddings, a more robust attention mechanism, and an enhanced training algorithm. These improvements have enabled the model to set new benchmarks in multiple NLP tasks.

The pre-trained Microsoft DeBERTa-v3-base model, just like BERT model, can be employed for various NLP applications such as text classification, question-answering, and text generation tasks. The pre-trained DeBERTa-v3-base model is provided by Hugging Face library with an easy-to-use interface which can be utilized in different programming languages. In this study, the pretrained DeBERTa-v3-base was finely tuned for sentiment classification of Covid-19 twitter dataset.

#### **4.3.2.4 MiniLM-L12-H384-uncased**

The pre-trained MiniLM-L12-H384-uncased language model is built on the original MiniLM model's [36] architecture. The small and more efficient variation of the original BERT model is the MiniLM model which achieves better performance with few parameters. The MiniLM-L12-H384-uncased model is a relatively small model as compared to other pre-trained language models as it consists of 12 transformer layers and a hidden size 384. The "uncased" means that the model doesn't distinguish between the upper-case and lower-case letters in the input text. This is helpful in situations where the case of the text is unimportant, or when the text involved isn't consistently cased.

The MiniLM-L12-H384-uncased model, trained on a large corpus of textual data including Wikipedia and web pages, has undergone masked language modeling (MLM) training. This model can be further fine-tuned for various NLP tasks such as text classification, question-answering, and named entity recognition. Despite its smaller size, the MiniLM-L12-H384-uncased model has demonstrated excellent performance in different benchmarks and exhibits faster and more efficient computation compared to larger models like BERT or GPT. In this particular study, the MiniLM-L12-H384-uncased model has been fine-tuned to classify the sentiment of Twitter tweets related to Covid-19 into five distinct classes.

#### **4.3.2.5 Twitter RoBERTa Base**

A new pre-trained model "Robustly optimized BERT approach (RoBERTa)" [37] was developed by Facebook AI Research (FAIR) in 2019. RoBERTa is the enhanced version of BERT model that can evaluate large and dynamic data sets, but the architecture of the model is same as BERT original. The RoBERTa model diverges with BERT model in its bigger vocabulary size, larger training data, longer training time which increases the pre-training optimization. Moreover, in pre-training RoBERTa employs dynamic token masking, which means tokens are randomly masked on each epoch training instead of a static scheduled masking to enhance the model's robustness and generalization. The performance of Roberta has been observed better in different benchmarks.

Twitter Roberta base model [38], built on original BERT architecture, is a variant of the original Roberta model series that has been optimized for language processing specifically on Twitter data. It has been trained on around 58M Twitter corpus which includes tweets and



other content from the platform and is specifically designed to understand and analyze the Twitter language. Fine tuning the model can perform various NLP tasks on Twitter data, such as named entity recognition, question-answering, and sentiment analysis. In this research, twitter-roberta-base model is used classify the tweet's sentiment into positive, extremely positive, negative, extremely negative and neutral classes. Multiple benchmarks and evaluations have acknowledged the performance of the twitter-roberta-base model. Thus, it has become popular for research on Twitter for NLP applications, including social media analysis, opinion mining and content moderation.

#### **4.3.2.6 Twitter RoBERTa Base Sentiment**

RoBERTa's base Twitter sentiment model [38] is a pre-trained language model based on the RoBERTa architecture. It was specially tuned for sentiment analysis tasks on Twitter data. The model was trained on a large corpus of Twitter data and can be further fine-tuned for specific sentiment-related tasks using labeled datasets. It is suitable for various NLP tasks involving Twitter sentiment analysis, such as brand monitoring, social media content analysis, and customer feedback evaluation. The model can be easily integrated into existing NLP workflows. It is trained to classify tweets into categories of negative, positive or neutral sentiment. The evaluation conducted on multiple benchmarks demonstrates that the model surpasses other sentiment analysis models trained specifically on Twitter data, achieving higher accuracy and F1 scores.

#### **4.3.2.7 Twitter RoBERTa Base 2022-154M**

The pre-trained Twitter RoBERTa base 2022 model is basically the original RoBERTa base model architecture which has been specifically trained on a tweet dataset consisting of 154M tweets, till the December end 2022. This model involved additional training to increase its performance as compared to the RoBERTa base model [37] and has more parameters i.e., large Twitter corpus data. The dataset was obtained by filtering 220M tweets entirely from the Twitter Academic API, containing the tweets between the months of January 2018 and December 2022.

The model, which has been fine-tuned for various natural language processing tasks on Twitter data, including sentiment analysis, named entity recognition, and question-answering, has undergone rigorous evaluation on multiple benchmarks. The results of these evaluations

have indicated that the model outperforms previous Twitter language-based models, achieving higher accuracy and F1-score metrics. The versatility of the model allows it to be utilized in a wide range of NLP tasks related to Twitter sentiment analysis, such as brand monitoring, social media content analysis, and customer feedback evaluation. Being an open-source model, it can be easily integrated into existing NLP workflows. In the present study, the model from the Hugging Face library was employed to classify Twitter tweets related to the Covid19 pandemic into five sentiment categories: positive, negative, extremely positive, extremely negative, and neutral.

## **Summary**

This part covers the detailed methodology for the sentiment analysis of Covid-19 tweets using the deep learning approach. The methodology consists of multiple phases i.e., pre-processing, creation of embedded vectors and sentimental classification of Twitter tweets related to Covid-19. The tweets' pre-processing part is covered at first followed by creating word vectors for feature selection and then the classification methodology has been discussed. The section also presents a detailed insights into the different variants of deep learning model from transformer BERT employed in the study for Covid-19 tweets sentiment classification.

## CHAPTER 5: EXPERIMENT & RESULTS

### 5.1 Dataset

#### 5.1.1 Realtime Twitter Dataset

The dataset used in this research consists of tweets collected from the Twitter platform related to the Covid-19 pandemic. To collect the tweets, a Twitter account was registered, and the necessary credentials (consumer keys, secret codes, access tokens, and access token secrets) were obtained from apps.twitter.com. These credentials were then used in an authentication routine to access Twitter data. The Tweepy Python library, which allows streaming of Twitter data, was utilized to stream the tweets. In order to obtain tweets from the peak period of the Covid-19 pandemic, the Harvard Dataverse Coronavirus Dataset [42] was utilized. This dataset contained tweet IDs but not the actual tweet content. By using these tweet IDs, tweets were collected between March 3, 2020, and December 3, 2020. The searchTwitter function was employed to extract tweets in English without retweets, using keywords such as #Coronavirus, #Coronaoutbreak, and #COVID19. Approximately 58,863 tweets related to Covid-19 were retrieved. Table 6 provides a sample of randomly selected tweets from the real-time Twitter dataset.

**Table 6: Randomly Selected Tweets from Real Time Twitter Dataset**

Tweet ID's	Tweet
1234912380867317760	#voteforanunaki Sounds like Trump's tryin' to keep #Coronavirus numbers down by refusin' to test folks. This is gonna be a total disaster
1234894096864595974	<a href="https://t.co/ECN63NIQMj">https://t.co/ECN63NIQMj</a> #CoronaOutbreak   In an attempt to contain the further spread of coronavirus cases in India, the central government on Tuesday cancelled all visa/eVisa issued to Italy, Iran, South Korea and Japan nationals on or before March 3
1234865364451971072	<a href="https://t.co/v1DvE4fCTx">https://t.co/v1DvE4fCTx</a> @educationgovuk has launched a new helpline to answer questions about #COVID19 related to education.
1234873582544658433	See our #Coronavirus guidance for nurseries here: <a href="https://t.co/nHsM71LNUu">https://t.co/nHsM71LNUu</a>

	<a href="https://t.co/PViBu9oEO5">https://t.co/PViBu9oEO5</a> From LAX, talking about #masks and the #coronavirus. To panic or not to panic, that is the question! #DontPanic! Don't believe the hysteria from @CNN and the #fakenews! Thank you President @realDonaldTrump or doing an amazing job!
1234823166926708741	Advising people to shake hands with people infected with #Covid_19 patients is mad & bad. <a href="https://t.co/P07xTzFEh6">https://t.co/P07xTzFEh6</a> With two cases of #COVID19 now confirmed in Georgia, I wanted to share what I learned in my briefing with @CDCgov yesterday.

### 5.1.2 Kaggle Corona NLP - Text Classification Dataset

The “Coronavirus tweets NLP - Text Classification” dataset used in this research is an open-source dataset obtained from Kaggle [39]. The dataset consists of the tweets related to Covid-19 collected using the search keywords as CORONA, COVID, COVID19, Covid 19, etc. The dataset contains around 44955 tweet records with five attributes and one Sentiment label. Information like username, screen name, user's location, tweet date, actual tweet text, and the user's sentiments or emotions are composed in the dataset. Table 7 shows the summary of dataset attributes. To maintain the privacy of the users, names and usernames have been replaced with codes. Sentiments have been classified into five types i.e., Positive, Negative, Extremely Positive, Extremely Negative and Neutral.

**Table 7: List of Dataset Attributes**

Attribute	Description
UserName	Name of the Twitter users
ScreenName	Display name of the Twitter users
Location	Location where the tweet has been posted
TweetAt	Time when the tweet has been posted
OriginalTweet	Text of the tweet been posted
Sentiment	Categorical value of classes

The distribution of the classes w.r.t their frequency in corpus is shown in figure 11. The granularity of the dataset is shown through the Word Cloud which is presented in figure 12. The term “HTTPS” has occurred frequently in the dataset, but it only refers to the links or URLs shared by the users. The URLs don’t provide any relevant information to sentiments. Such challenges in raw tweets have been handled in the preprocessing phase for enhancing the model’s effectiveness. Table 8 shows randomly selected tweets from the dataset for each class.

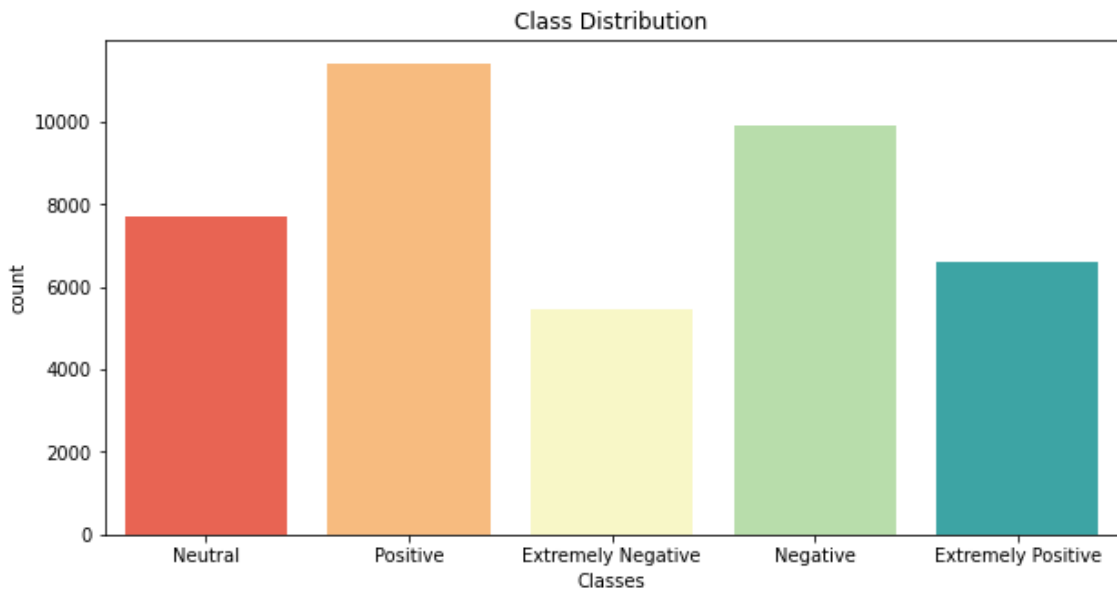


Figure 11: Class Distribution by Label



Figure 12: Word Cloud of Dataset

**Table 8: Randomly Selected Tweets from Kaggle Dataset for Each Class**

<b>Extremely Negative</b>	<b>Negative</b>	<b>Neutral</b>	<b>Positive</b>	<b>Extremely Positive</b>
<p>Farmers are plowing crops under because of no restaurant demand We see unimaginable lines of desperate citizens at food banks Why not use some of the Trillions being thrown around to buy the crops from the farmers and give it to the citizens in need pence</p>	<p>We're sorry to say that our @FinFabUK event is being cancelled due to Covid-19. The health and wellbeing of our attendees, speakers and staff is our top priority. Apologies for any disappointment this may cause. All FAQs are answered in the link below:<a href="https://t.co/GDDPTudCvj">https://t.co/GDDPTudCvj</a></p>	<p>@MeNyrbie @Phil_Gahan @Chrisity <a href="https://t.co/iFz9FAn2Pa">https://t.co/iFz9FAn2Pa</a> and <a href="https://t.co/xX6ghGFzCC">https://t.co/xX6ghGFzCC</a> and <a href="https://t.co/I2NlzdXNo8">https://t.co/I2NlzdXNo8</a></p>	<p>advice Talk to your neighbours family to exchange phone numbers create contact list with phone numbers of neighbours schools employer chemist GP set up online shopping accounts if poss adequate supplies of regular meds but not over order</p>	<p>Morning everyone have a great and safe day. ??? #coronavirus #StopPanicBuying #BeKind #mufc #MUFC_Family</p>
<p>@CNN People's food requirements haven't changed because of Covid-19; only their buying habits. If people would stop panic buying and hoarding there would be no empty shelves at the stores. People need to stop panicking and stop being selfish!</p>	<p>Why we stock up on water... cause utility companies will shut you off in the middle of a pandemic... the schools close thier doors, you lose out on work cause your kid has no where to go... and you can't afford months worth of food. #coronavirus @SenatorRomney <a href="https://t.co/0CV0793olS">https://t.co/0CV0793olS</a></p>	<p>This is the line outside @Target in as customers wait for the store to open this morning</p>	<p>Coronavirus Australia: Woolworths to give elderly, disabled dedicated shopping hours amid COVID-19 outbreak <a href="https://t.co/bInCA9Vp8P">https://t.co/bInCA9Vp8P</a></p>	<p>Just called mum and dad in UK (over 70). They are great but I offered help with online shopping etc. We might sometimes forget that this is not always easy.Do the same if you can. ??If you are far from your parents like me Tech can be really useful. #COVID?19 #Coronavirus</p>

CHECK VIDEO ?? <a href="https://t.co/1ksn9Brl02">https://t.co/1ksn9Brl02</a> ??No food ? in USA market due to coronavirus panic we gonna die from starvation #CoronavirusOut break #coronavirus #houston #nofood #Notoiletpaper #NoHandShakes #nohandsanitizer #COVID19 #pandemic #totallockdown #COVID2019usa #walmart <a href="https://t.co/ztN3iMkgpD">https://t.co/ztN3iMkgpD</a>	"Consumer Corner: #Scammers Taking Advantage Of #COVID-19 Fears #coronavirus #cdc #flu #trends #alert <a href="https://t.co/sk9qCJsnY1">https://t.co/sk9qCJsnY1</a> <a href="https://t.co/T7qejP3hys">https://t.co/T7qejP3hys</a> "	"Electric Car News: Why Tesla Stock Jumped on Monday #News" #StayHomeSaveLives: <a href="https://t.co/RDHC32onfm">https://t.co/RDHC32onfm</a>	Worried about the impact of the current COVID-19 pandemic on your finances? We've just published some tips to help you manage your money during these challenging times. #COVID19 <a href="https://t.co/3jKK3CqXfQ">https://t.co/3jKK3CqXfQ</a> <a href="https://t.co/EbEnURmmJS">https://t.co/EbEnURmmJS</a>	"THANK YOU To all the grocery store employees for working so hard making sure everyone is getting what they need. Please be kind to them it's not their fault that we are short on supplies. #Corona #covid_19 @... <a href="https://t.co/QC0uNeVQejj">https://t.co/QC0uNeVQejj</a> "
---	--	--	---	--

### 5.1.3 Train-Test Split

The train-test split is a widely used technique in machine learning to assess the performance of a model on unseen data. It involves dividing the dataset into two subsets: a training set and a testing set. The training set is used to train the model, while the testing set is used to evaluate its performance on unseen data. The split ratio between the training and testing sets can vary depending on factors such as the size of the dataset and the complexity of the model. Typically, a larger portion of the dataset is allocated to the training set, while a smaller portion is reserved for testing. This ensures that the model learns from a substantial amount of data and is then evaluated on a separate, independent subset. In this study, dataset hasn't been manually split using train-test split technique since the dataset "Coronavirus tweets NLP - Text Classification" provided on Kaggle is already split into train and test sets, provided as two separate data files. The train set contains 41157 tweet records whereas the test set

contains 3798 tweet records. Tables 9 and 10 show the train and test set split w.r.t each sentiment class.

**Table 9: Train Dataset Count for Each Class**

Sentiment Class	Counts
Positive	11422
Negative	9917
Neutral	7713
Extremely Positive	6624
Extremely Negative	5481

**Table 10: Test Dataset Count for Each Class**

Sentiment Class	Counts
Positive	947
Negative	1041
Neutral	619
Extremely Positive	599
Extremely Negative	592

During the training phase, the model is trained using the training data to adjust its parameters and learn patterns from the input data. Once the model is trained, it is then evaluated on the testing or unseen data in testing phase. It helps us assess the model's generalization ability and determine if it can effectively handle real-world scenarios beyond the training data.



## **5.2 Performance Metrics**

### **5.2.1 Accuracy/ Precision**

In data analysis and measurements, accuracy is a significant concept [25]. Accuracy implies the degree of closeness between a measured or predicted value and its true or actual value. Statistically, accuracy is computed as the proportion of true predictions or classifications to the total observations in the dataset. Precision measures the proportion of correctly predicted positive cases out of the total number of predicted positive cases.

### **5.2.2 Recall**

It measures model completeness by measuring the proportion of correctly predicted positive cases out of the total number of true positive cases.

### **5.2.3 F1-score**

The F1 score is a statistical measure that combines precision and recall providing a more comprehensive assessment of the performance of a classification model, especially in cases where it is important to account for misclassification. It is calculated as the harmonic mean of precision and recall.

### **5.2.4 Confusion Matrix**

A confusion matrix is a table used to evaluate the performance of a classification model by comparing predicted and actual values. It provides a summary of the predictions made by the model along with the actual results. A matrix is a 2D array with predicted values on one axis and actual values on the other axis. The entries in the matrix represent the number of times each predicted class was correct or incorrect based on the actual results. The four possible outcomes are:

- True Positive (TP): the model correctly predicts the positive results.
- False positive (FP): the model incorrectly predicts the positive results.
- True Negative (TN): the model correctly predicts the negative results.
- False negative (FN): the model incorrectly predicts the negative results.

The confusion matrix is used to calculate several metrics, including precision, accuracy, recall, F1 score, and more metrics. These metrics are calculated based on the values in the

matrix, like total true predictions, false positives, false negatives, and true negatives. Figure 13 shows the confusion matrix of the proposed BERT model on the test results.

<b>864</b>	2	23	23	45
1	<b>489</b>	1	52	0
68	1	<b>587</b>	62	1
19	54	27	<b>784</b>	2
82	1	0	1	<b>566</b>

**Figure 13: Confusion Matrix of the Proposed BERT Test Results**

### 5.2.5 Experimental Setup & Parameters involved in Models Training

All the variants of Bert model were imported from HuggingFace library. All the models were run on an A100 GPU and PyTorch version 2.0.1+cu118 with 32GB of memory. The performance of the model relies on how the model has been trained using the most optimum hyperparameters. A few parameters used to train the models and their fine-tuned values have been listed in table 11 below.

**Table 11: Fine-tuned Hyperparameters Involved in Models' Training**

<b>Models</b>	<b>Epochs</b>	<b>Learning Rate</b>	<b>Batch Size</b>	<b>Number of Labels</b>
Random Forest				5
Distil BERT	10	0.00005	8	5
BERT	10	0.000001	32	5
microsoft/deberta-v3-base	25	0.00005	6	5
microsoft/MiniLM-L12-H384-uncased	15	0.00005	6	5
cardiffnlp/twitter-roberta-base	15	0.00005	8	5
cardiffnlp/twitter-roberta-base-sentiment	15	0.00005	8	5
cardiffnlp/twitter-roberta-base-2022-154m	15	0.00005	8	5

### 5.3 Experimental Results of BERT with Multiple Hyperparameters

The performance of BERT model is tested with different hyperparameters and the most optimum of the hyperparameter values are used. A few experiments with the following hyperparameters are performed to evaluate the effectiveness of the BERT model.

#### 5.3.1 Effects of Batch Size Number on the Accuracy of BERT

A few experiments with a different number of batch sizes of 32 and 64 have been executed. Initially batch size was selected as 32 and then it was increased to 64. The best accuracy attained was with 32 batch size, with less validation loss. On increasing the batch size, accuracy remained constant, but validation loss increased. The train accuracy and test accuracy values observed for all batch sizes are mentioned in Table 12.

**Table 12: Effects of Batch Size Number on the Accuracy of BERT**

<b>(Number of Epochs= 10, Test Data Size=0.1)</b>				
<b>Batch Size</b>	<b>Training Loss</b>	<b>Validation Loss</b>	<b>Train Accuracy</b>	<b>Test Accuracy</b>
32	0.04	1.02	88%	88%
64	0.04	1.05	88%	88%

#### 5.3.2 Effects of Epochs Number on the Accuracy of BERT

A few experiments have been performed with different epoch numbers as 10 and 20. At first, the epoch number was set as 10 and then increased to 20. The highest accuracy was observed at epoch 10 and then on increasing the epochs number the accuracy remained constant, but the validation loss increased. So, the number of epochs as 10 was selected as best parameter value. The train accuracy and test accuracy values observed for every epoch are mentioned in Table 13.

**Table 13: Effects of Epochs Number on the Accuracy of BERT**

<b>(Batch Size= 32, Test Data Size=0.1)</b>				
<b>No. of Epochs</b>	<b>Training Loss</b>	<b>Validation Loss</b>	<b>Train Accuracy</b>	<b>Test Accuracy</b>
10	0.04	1.02	88%	88%
20	0.005	1.31	88%	88%

**5.3.3 Effects of Learning Rate on the Accuracy of BERT**

Experiments with a different number of learning rates as  $1e^{-3}$ ,  $1e^{-5}$ ,  $2e^{-5}$  and  $3e^{-5}$  have been performed. Initially the learning rate value was set as default i.e.,  $1e^{-3}$  and then it was gradually increased. The highest accuracy was obtained with the learning rate  $1e^{-5}$  with least validation loss whereas on increasing the learning rate value the accuracy remained constant, but validation loss increased. The train accuracy and test accuracy values observed for every learning rate are mentioned in Table 14.

**Table 14: Effects of Learning Rate on the Accuracy of BERT**

<b>(Batch Size=32, Test Data Size=0.1)</b>				
<b>Learning Rate</b>	<b>Training Loss</b>	<b>Validation Loss</b>	<b>Train Accuracy</b>	<b>Test Accuracy</b>
$1e^{-3}$ (Default)	1.58	1.58	12%	28%
$1e^{-5}$	0.04	1.02	88%	88%
$2e^{-5}$	0.017	1.07	88%	88%
$3e^{-5}$	0.03	1.04	88%	87%

## 5.4 Experimental Results of BERT

Once the BERT model was trained with the best optimum selected hyperparameters, the model was tested with two different datasets i.e., one with Kaggle Corona NLP test dataset and second Custom Real Time Twitter Dataset. The following table 15 shows the evaluation of applied BERT model on Kaggle test dataset with respect to each sentiment class. The results of the proposed Bert model on the real time custom Twitter dataset are presented in table 16.

**Table 15: Experimental Results of BERT model on Test Dataset**

<b>Class</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>
Overall Accuracy			<b>88%</b>
Extremely Negative (0)	0.84	0.90	0.87
Positive (1)	0.89	0.90	0.90
Extremely Positive (2)	0.92	0.82	0.87
Negative (3)	0.85	0.88	0.87
Neutral (4)	0.92	0.87	0.90

**Table 16: Experimental Results of BERT model on Custom Twitter Dataset**

<b>Tweets Text</b>	<b>Predicted Sentiment</b>	<b>Actual Sentiment</b>
Since this will never get reported by the media, I wanted to share a copy of this check. @realDonaldTrump is once again donating his salary back to the United States Government — This quarter, it will be donated to @HHSGov to confront, contain, and combat #Coronavirus. usus <a href="https://t.co/hVZsm6z1zu">https://t.co/hVZsm6z1zu</a>	Extremely Negative	Positive
LIVE daily media briefing on #COVID19 with @DrTedros #coronavirus <a href="https://t.co/Uuup642t9d">https://t.co/Uuup642t9d</a>	Negative	Negative
Coronavirus: Nasa images show China #POLLUTION clear amid slowdown <a href="https://t.co/3T4K8An7qS">https://t.co/3T4K8An7qS</a> #GPWX	Extremely Positive	Extremely Positive
WHO calls on industry and governments to increase manufacturing of personal protective equipment by 40% to meet rising global demand due to #COVID19 👉 <a href="https://t.co/XM7RlcivuV">https://t.co/XM7RlcivuV</a> #coronavirus <a href="https://t.co/c5JTXdpQm7">https://t.co/c5JTXdpQm7</a>	Neutral	Neutral
Coronavirus now appears to be spreading much more rapidly outside China than within but can still be contained, and stigma is more dangerous than the disease itself, the World Health Organization says <a href="https://t.co/Lxs2QZLkJt">https://t.co/Lxs2QZLkJt</a> <a href="https://t.co/r0gIAIR17k">https://t.co/r0gIAIR17k</a>	Neutral	Extremely Negative

## 5.5 Comparison of the Models on Twitter Sentiment Analysis

A number of BERT variants were fine-tuned for analyzing the Covid-19 tweets' sentiment with optimum selected hyperparameters and then the results of the BERT model were compared with the other models. Table 17 shows the comparison of all models for the covid sentiment analysis.

**Table 17: Comparison of the Models on Twitter Sentiment Analysis**

Frameworks	Models	Accuracy
Framework1	Random Forest	0.63
Framework 2	Distil BERT	0.87
	BERT	<u>0.88</u>
	microsoft/deberta-v3-base	0.25
	<b>microsoft/MiniLM-L12-H384-uncased</b>	<b>0.93</b>
	cardiffnlp/twitter-roberta-base	0.85
	cardiffnlp/twitter-roberta-base-sentiment	0.86
	cardiffnlp/twitter-roberta-base-2022-154m	0.85

## 5.6 Comparison with Existing Techniques

The results of the proposed model were compared with the existing BERT sentiment models. The results in table 18 shows that our proposed Bert model performed better in accuracy with less number of training time

**Table 18: Comparison of the Model with Existing Techniques**

Technique	Accuracy
Rifat et al. [24]	87%
Michal Ashkenazi [42]	70.5%
Edgard Jonathan [43]	86%
<b>Proposed Model</b>	<b>88%</b>

## Summary

This section provides detailed insights into the two datasets i.e., Kaggle NLP Corona Dataset and Realtime Custom Twitter Dataset employed in the study. A detailed ablation study was conducted to experiment with different available models. The experimental results obtained from applying the Machine Learning (Random Forest) and multiple Deep Learning models for sentiment analysis of Twitter tweets related to the Covid-19 pandemic have been discussed and then in the last a comparison of all the models' results has been presented in this section.

## CHAPTER 6: CONCLUSION

We conducted a study to examine public attitudes and perceptions towards the Coronavirus and COVID-19 epidemic, which revealed an increase in negative attitudes and feelings of dread. To achieve this, we utilized Twitter as a valuable source of information for sentiment analysis. We evaluated the performance of a machine learning model i.e., Random Forest, along with eighth deep learning models including BERT. Among the deep learning models, BERT, which is a bidirectional open-source NLP model developed by Google. The deep learning models include BERT base, Distil-BERT, DeBERTa base, MiniLM, Twitter RoBERTa base, Twitter RoBERTa base sentiment, and Twitter RoBERTa base 2022. BERT is unique for its self-attention mechanism. We compared the performance of these models for sentiment analysis on Twitter. The results proved that DL models perform well as compared to ML models in which BERT base and MiniLM exceeds the performance by achieving 88% and 93% accuracy respectively. This is attributed to Simple BERT's utilization of the Bidirectional Transformer with MLM and NSP, as well as being trained on Book corpus and Wikipedia, with a total of 3.3 billion words. We assume that MiniLM utilizes the Bert Transformer architecture, so it outperforms the BERT model. Additionally, MiniLM is the improved and enhanced smaller version of BERT and has been pre-trained on more diverse data than BERT, which improves the generalization.

To improve the analysis of public sentiment on crucial topics like government response to a pandemic, government healthcare facilities, offline inspection, and mental health, deep learning (DL) algorithms are likely to be used more frequently in the future. However, deep data analysis revealed that the assignment of sentiment to the tweets isn't very correct, so further investigation and dedicated effort is required to further enhance the quality of the labels.



## REFERENCES

- [1] Sharma, A. and Daniels, A., 2020, "Tweets Sentiment Analysis via Word Embeddings and Machine Learning Techniques."
- [2] Rezaeinia, S., Ghodsi, A. and Rahmani, R., 2017, "Improving the Accuracy of Pre-trained Word Embeddings for Sentiment Analysis."
- [3] B. Oscar Deho, A. William Agangiba, L. Felix Aryeh and A. Jeffery Ansah, 2018, "Sentiment Analysis with Word Embedding," *2018 IEEE 7th International Conference on Adaptive Science & Technology (ICAST)*, pp. 1-4
- [4] A. N. Farhan and M. L. Khodra, 2017, "Sentiment-specific word embedding for Indonesian sentiment analysis," *2017 International Conference on Advanced Informatics, Concepts, Theory, and Applications (ICAICTA)*, pp. 1-5.
- [5] H. Imaduddin, Widyawan and S. Fauziati, 2019, "Word Embedding Comparison for Indonesian Language Sentiment Analysis," *2019 International Conference of Artificial Intelligence and Information Technology (ICAIT)*, pp. 426-430.
- [6] M. Al-Amin, M. S. Islam and S. Das Uzzal, 2017, "Sentiment analysis of Bengali comments with Word2Vec and sentiment information of words," *2017 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pp. 186-190.
- [7] W. Yue and L. Li, 2020, "Sentiment Analysis using Word2vec-CNN-BiLSTM Classification," *2020 Seventh International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pp. 1-5.
- [8] Q. Li, S. Shah, R. Fang, A. Nourbakhsh and X. Liu, 2016, "Tweet Sentiment Analysis by Incorporating Sentiment-Specific Word Embedding and Weighted Text Features," *2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 568-571.
- [9] Y. Sharma, G. Agrawal, P. Jain and T. Kumar, 2017, "Vector representation of words for sentiment analysis using GloVe," *2017 International Conference on Intelligent Communication and Computational Techniques (ICCT)*, pp. 279-284.
- [10] Ren, Y., Wang, R. and Ji, D., 2016. "A topic-enhanced word embedding for Twitter sentiment classification," *2016 Information Sciences*, 369, pp.188-198.
- [11] Sitaula, C., Basnet, A., Mainali, A. and Shahi, T., 2021, "Deep Learning-Based Methods for Sentiment Analysis on Nepali COVID-19-Related Tweets," *2021 Computational Intelligence and Neuroscience*.

- [12] Shofiya, C. and Abidi, S., 2021. "Sentiment Analysis on COVID-19-Related Social Distancing in Canada Using Twitter Data," 2021 *International Journal of Environmental Research and Public Health*, 18(11), p.5993.
- [13] Abdulaziz, M., Alotaibi, A., Alsolamy, M. and Alabbas, A., 2021, "Topic based Sentiment Analysis for COVID-19 Tweets," 2021 *International Journal of Advanced Computer Science and Applications*, 12(1).
- [14] Rehman, A., Malik, A., Raza, B. and Ali, W., 2022, "A Hybrid CNN-LSTM Model for Improving Accuracy of Movie Reviews Sentiment Analysis," 2022 *Multimed Tools Applications*, 78, pp. 26597–26613.
- [15] Pak, A. and Paroubek, P., 2010, "Twitter as a Corpus for Sentiment Analysis and Opinion Mining," 2010 *Proceedings of LREC*. 10.
- [16] Kaur, C. and Sharma, A., 2020, "Twitter Sentiment Analysis on Coronavirus using Textblob," 2020 *EasyChair*, No. 2974.
- [17] Ravindran, S. K., & Garg, V., 2015, "Mastering social media mining with R," 2015 *Packt Publishing Ltd*.
- [18] I. Guellil and K. Boukhalfa, 2015, "Social big data mining: A survey focused on opinion mining and sentiments analysis," 2015 *12th International Symposium on Programming and Systems (ISPS)*, pp. 1-10
- [19] Twitter [Online] <https://backlinko.com/twitter-users>
- [20] Fung, I., Duke, C., Finch, K., Snook, K., Tseng, P., Hernandez, A., Gambhir, M., Fu, K. and Tse, Z., 2016, "Ebola virus disease and social media: A systematic review," 2016 *American Journal of Infection Control*, Vol. 44, Issue 12, pp.1660-1671.
- [21] Rasool, A., Tao, R., Marjan, K. and Naveed, T., 2019, "Twitter Sentiment Analysis: A Case Study for Apparel Brands," 2019 *Journal of Physics: Conference Series*, Vol. 1176, No. 2, p.022015.
- [22] S. Shayaa, P. S. Wai, Y. W. Chung, A. Sulaiman, N. I. Jaafar, and S. B. Zakaria, 2017, "Social Media Sentiment Analysis on Employment in Malaysia", 2017 *Proceedings of 8th Global Business and Finance Research Conference*, Taipei, Taiwan.
- [23] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," pp. 1–12, 2013
- [24] N. Rifat, M. Ahsan, R. Gomes and M. Chowdhury, "COVID-19 Sentiment Analysis applying BERT," 2022 *IEEE International Conference on Electro Information Technology (eIT)*, 2022, pp. 417-422.

- [25] K. Mahor and A. K. Manjhvar, "Public Sentiment Assessment of Coronavirus-Specific Tweets using a Transformer-based BERT Classifier," 2022 International Conference on Edge Computing and Applications (ICECAA), 2022, pp. 1559-1564
- [26] T. S. Sai Kumar, K. Arunagiri Pandian, S. Thabasum Aara and K. Nagendra Pandian, "A Reliable Technique for Sentiment Analysis on Tweets via Machine Learning and BERT," 2021 Asian Conference on Innovation in Technology (ASIANCON), 2021, pp. 1-5.
- [27] A. Topbaş, A. Jamil, A. A. Hameed, S. M. Ali, S. Bazai and S. A. Shah, "Sentiment Analysis for COVID-19 Tweets Using Recurrent Neural Network (RNN) and Bidirectional Encoder Representations (BERT) Models," 2021 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube), 2021, pp. 1-6.
- [28] A. J. Nair, V. G and A. Vinayak, "Comparative study of Twitter Sentiment On COVID - 19 Tweets," 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), 2021, pp. 1773-1778.
- [29] N. Chintalapudi, G. Battineni, and F. Amenta, "Sentimental Analysis of COVID-19 Tweets Using Deep Learning Models," *Infectious Disease Reports*, vol. 13, no. 2, pp. 329–339, Apr. 2021
- [30] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, 2019 "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Proceedings of the 2019 Conference of the North*, May 2019.
- [31] Sadia, K. and Basak, S., 2021, 'Sentiment analysis of covid-19 tweets: How does Bert perform?', *2021 Algorithms for Intelligent Systems*, pp. 407–416.
- [32] Singh, D., 2022, "Analysis Of Public Sentiment of Covid-19 Pandemic, Vaccines, And Lockdowns".
- [33] HuggingFace [Online] <https://huggingface.co/bert-base-uncased>
- [34] Sanh, V., Debut, L., Chaumond, J. and Wolf, T., 2019, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter,".
- [35] He, P., Gao, J., & Chen, W., 2021, Deberv3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing.
- [36] Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou, M., "Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers", *Advances in Neural Information Processing Systems*, 33, 5776-5788, 2020

- [37] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, M. & Stoyanov, V., “Roberta: A robustly optimized bert pretraining approach”, 2019.
- [38] Barbieri, F., Camacho-Collados, J., Neves, L., & Espinosa-Anke, L., “Tweeteval: Unified benchmark and comparative evaluation for tweet classification”, *ACL Anthology*, 2020.
- [39] Kaggle Dataset [Online] <https://www.kaggle.com/datasets/datatattle/covid-19-nlp-text-classification>
- [40] Dussa, A., “Finetuning Pre-trained language models for sentiment classification of COVID19 tweets”, 2020.
- [41] Jangir, S., “Finetuning BERT and XLNet for Sentiment Analysis of Stock Market Tweets using Mixout and Dropout Regularization”, 2021.
- [42] Harvard Dataverse Dataset [Online] <https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/LW0BTB&version=9.1>
- [42] Kaggle Notebook [Online]: <https://www.kaggle.com/code/michalashkenazi/covid19-text-classification-nlp-bert-tensorflow/notebook>
- [43] Kaggle Notebook [Online]: <https://www.kaggle.com/code/edgardjonathan/bert-deep-learning>