TM-BERT: Sentiment Analysis on Covid Vaccination Tweets using Twitter Modified BERT

Author

Muhammad Talha Riaz

Reg. Number

00000275099

Supervisor

Dr. Muhammad Usman Akram

DEPARTMENT OF SOFTWARE ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

April, 2022

TM-BERT: Sentiment Analysis on Covid Vaccination Tweets using Twitter Modified BERT

Author
Muhammad Talha Riaz
Reg. Number
00000275099

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 SOFTWARE ENGINEERING
COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING
NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY,
ISLAMABAD
April, 2022

DECLARATION

I declare that this research work titled "*TM-BERT: Sentiment Analysis on Covid Vaccination Tweets using Twitter Modified BERT"* is my own work. The work has not been demonstrated somewhere else for appraisal. The used material which has been taken from other sources has been correctly acknowledged/referred.

Signature of Student
Muhammad Talha Riaz
Reg. Number
00000275099

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
Muhammad Talha Riaz
Reg. Number
00000275099

Signature of Supervisor
Dr.Muhammad Usman Akram

Copyright Statement

Acknowledgments

All the glorification and praise to Almighty ALLAH (the most glorified, the highest) who gave me the courage, tolerance, knowledge, familiarity and competence to carry out this work and to endure and finish it satisfactorily. Undoubtedly, HE soothes my way and without HIS blessings I can’t achieve anything.

I would like to express my truthful gratitude to my advisor Dr. Muhammad Usman Akram for boosting my morale and for his constant support, encouragement, devotion and precious supervision in my quest for knowledge. I am blessed to have such a cooperative advisor and generous mentor for my research.

Along with my advisor, I would like to acknowledge my entire thesis committee: Dr. Sajid Gul Khawaja and Dr. Arslan Shaukat for their assistance and judicious recommendations.

My acknowledgment would be incomplete without thanking the biggest source of my strength, my family. I am generously grateful to my beloved parents 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 and sisters who were with me through my thick and thin.

Lastly, I would like to express my appreciation to all my friends particularly Muhammad Shah Jahan and the individuals who have encouraged and supported me through this entire period.

Dedicated to my Late father, supportive mother, brother and sisters whose incredible encouragement and support led me to this accomplishment

Abstract

Sentiment Analysis is an ongoing field of research in Natural Language Processing (NLP) particularly aimed at analyzing subjective and textual information to extract judgmental or behavioral knowledge for the computational treatment of opinions and sentiments of individuals. In transfer learning a model is pre-trained on a large unsupervised dataset and then fine-tuned on domain-specific downstream tasks. BERT is the first true-natured deep bidirectional language model which reads the input from both sides of input to better understand the context of a sentence by solely relying on the Attention mechanism. This study presents a Twitter Modified BERT (TM-BERT) based upon Transformer architecture. It has also developed a new Covid-19 Vaccination Sentiment Analysis Task (CV-SAT) and a COVID-19 unsupervised pre-training dataset containing (70K) tweets. BERT achieved (0.70) and (0.76) accuracy when fine-tuned on CV-SAT, whereas TM-BERT achieved (0.89), a (19%) and (13%) accuracy over BERT. Another enhancement introduced is in terms of time efficiency as BERT takes (64) hours of pre-training while TM-BERT takes only (17) hours and still produces (19%) improvement even after pre-trained on four (4) times fewer data.

Key Words: *BERT, TM-BERT, bidirectional language modeling, Covid Vaccination, Covid-19, Natural Language Processing (NLP).*

TABLE OF CONTENTS

DECLARATION ..... **I**

LANGUAGE CORRECTNESS CERTIFICATE ..... **II**

COPYRIGHT STATEMENT ..... **III**

ACKNOWLEDGMENTS ..... **IV**

ABSTRACT ..... **VI**

CHAPTER 1: INTRODUCTION **2**

1.1 BACKGROUND & MOTIVATION 4

1.2 PROBLEM STATEMENT 7

1.3 AIMS AND OBJECTIVES 7

1.4 STRUCTURE OF THE THESIS 7

CHAPTER 2: LANGUAGE MODEL **9**

2.1 LANGUAGE MODEL 9

2.2 HOW LANGUAGE MODELS WORK 9

2.3 TYPES OF LANGUAGE MODELS 10

2.3.1 N-gram 10

2.3.2 Unigram 10

2.3.3 Bidirectional 11

2.3.4 Exponential 11

2.4 PRE-TRAINING 12

2.4.1 Supervised Pre-training 13

2.4.2 Unsupervised 13

2.4.3 Semi-supervised 13

2.5 FINE-TUNING 13

2.6 SUMMARY 14

CHAPTER 3: LITERATURE REVIEW **17**

3.1 BERT TRAINED WITH DIFFERENT STRATEGY 18

3.2 RELATED WORK 25

3.3 LITERATURE SUMMARY 29

3.4 RESEARCH GAPS 30

CHAPTER 4: METHODOLOGY **32**

4.1 WORKING OF BERT 32

4.2 PRE-TRAINING 32
4.2.1 Token Embeddings 33
4.2.2 Segment Embeddings 33
4.3 FINE-TUNING 34
4.4 COVID-19 VACCINATION SENTIMENT ANALYSIS TASK (CV-SAT) 34
4.5 TWITTER UNSUPERVISED DATASET 37
4.6 TM-BERT MODEL 39
4.7 PERFORMANCE PARAMETERS 40
4.7.1 Vocabulary Size 40
4.7.2 Sequence Size 40
4.7.3 Batch Size 40
4.7.4 Step Size 41
4.7.5 Learning Rate 41
4.7.6 Hidden Layers 41
4.7.7 Input Layers 41
4.7.8 Attention Heads 41
4.7.9 Intermediate-Size 41
4.7.10 Maximum-Position-Embeddings 41
4.8 SUMMARY 41
CHAPTER 5: IMPLEMENTATION AND RESULTS 44
5.1 IMPLEMENTATION 44
5.2 PRE-TRAINING OF BERT 44
5.3 FINE-TUNING OF BERT 45
5.4 DATASET FOR PRE-TRAINING 45
5.5 PRE-TRAINING OF BERT_1 46
5.6 FINE-TUNING OF BERT_1 46
5.7 PRE-TRAINING OF TM-BERT 46
5.8 FINE-TUNING OF TM-BERT 48
5.9 RESULTS 49
5.10 SUMMARY 50
CHAPTER 6: DISCUSSION, CONCLUSION AND FUTURE WORK 52
CONTRIBUTIONS 53
REFERENCES 54

LIST OF FIGURES

Figure 1 Sentiment Analysis Flow....................2
Figure 2 Language Model Illustration....................10
Figure 3 Pre-training Illustration....................12
Figure 4 Fine-tuning Illustration....................14
Figure 5 Input Vector....................34
Figure 6 Flow Diagram of Data Collection....................35
Figure 7 Dev Sample....................35
Figure 8 Get_Test Sample....................36
Figure 9 Train Sample....................36
Figure 10 Unprocessed Data Sample....................37
Figure 11 Processed Data Sample....................38
Figure 12 Word Cloud Sample....................38
Figure 13 Statistical Analysis of Covid Vaccination Tweets....................39
Figure 14 Flow Diagram of TM-BERT Model....................40
Figure 15 TM-BERT Hyperparameters....................47
Figure 16 Hyper-parameters Comparison....................48
Figure 17 Final Results....................50

LIST OF TABLES

Table 1 Overall Result 22

Table 2 Overall Hyperparameters..24

Table 3 Overall Results..29

Table 4 Pre-training Hyper-parameters 45

Table 5 BERT_1 Pre-Training Dataset..46

Table 6 TM-BERT Pre-Training Dataset 46

Table 7 Hyper-parameters for BERT_1 pre-training on Wikipedia + Tweets 46

Table 8 Hyperparameters of TM-BERT 47

Table 9 All Pre-Trained Models 49

Chapter 1

Introduction

CHAPTER 1: INTRODUCTION

Sentiment Analysis, also known as Opinion Mining (OM) is an ongoing field of research in natural language processing (NLP) particularly aimed at analyzing subjective and textual information to extract evaluative, judgmental, or behavioral knowledge for the computational treatment of opinions and sentiments of individuals towards a particular subject or element[1] As the recurrence and relevance of user's opinions are considered the most vital success factor of sentiment analysis, Twitter has become the most useful and reliable resource for performing it because of its allowance to identify and collate all sentiments and opinions with ease through analyzing tweets, retweets and hashtags that individuals share. Since the COVID-19 outbreak, quarantine measures have been taking place across the globe for the sake of containment, restricting people to their places of shelter. In such dire times of global pandemic, Twitter proved to be a vital source for getting reliable information about the COVID-19 situation worldwide.

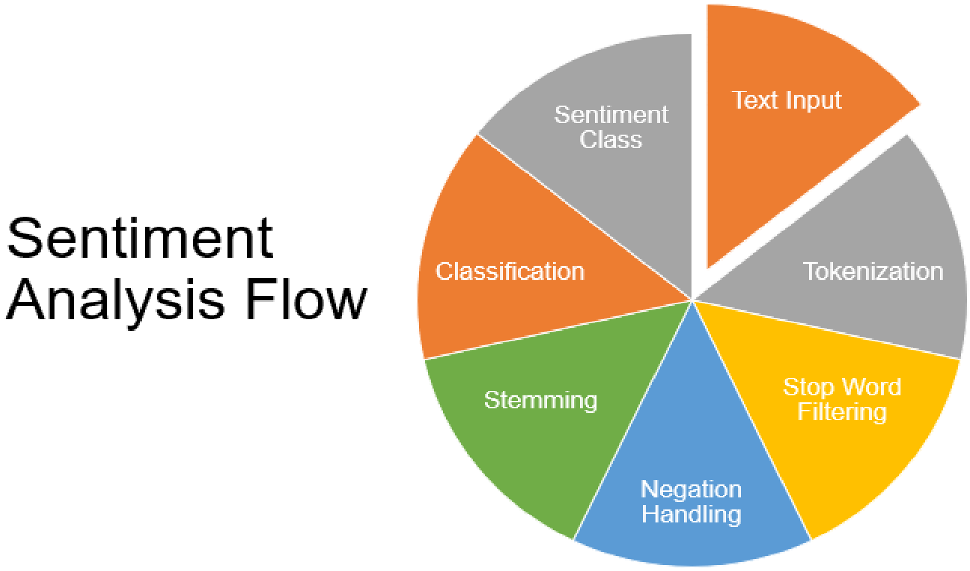


Figure 1 Sentiment Analysis Flow

In the days of lockdown, for providing the public of several key countries with the latest credible and authoritative information on COVID-19, Twitter added a specially dedicated tab in its 'Explore' which provided COVID-19 related public service announcements, tweets from health officials, suggested precautionary measures from health experts and the survival stories of COVID-19 infected people. Due to it, Twitter became a hub for billions of conversations related to the pandemic, ranging from valuable information to personal experiences, connecting people with the varying reaction towards COVID-19 across the globe. Many researchers conducted surveys and performed sentiment analysis on Twitter tweets to analyze people's behavior towards COVID-19 [2] but an inadequate focus has been given to the people's reaction specific towards COVID-19 vaccination. As COVID-19 vaccination was made mandatory for every individual to counter the virus, people from developed countries like the USA and the UK, as well as from developing countries like Pakistan, gave mixed reactions towards the vaccination. In a developing country especially Pakistan, where the literacy rate is low as compared to the developed countries, it is of vital importance to analyze the population's reaction towards COVID-19 Vaccination.

Sentiment analysis can be performed through various techniques including Machine Learning (ML), Information Theory and Coding (ITC), Decision Tree (DT), Semantic Orientation (SO), Natural Language Processing (NLP) as well as through Hybrid approaches too. Among all the approaches mentioned, NLP is considered the vastest field presenting multiple techniques for sentiment analysis. In Natural Language Processing tasks Language Models are used. Firstly, these models were trained on supervised data and then these trained models transfer knowledge to downstream tasks. Due to the limit of monitored data sets with human labels these models were trained on rarer data and produced limited results on NLP tasks such as questioning, classification, etc. Data is doubling up over time, resulting in a very giant unsupervised dataset so these Language Models started to train on these giant unsupervised datasets like WIKIPEDIA, BOOKSCORPUS, etc, and then fine-tuned on a supervised dataset. These new pretraining techniques reform transfer learning and produced SOTA results. These pre-trained models built on RNN architecture retain limited memory so when a sentence becomes larger the memory needed to maintain a relationship also becomes larger which makes the model more limited and less rapid. Transformer architecture reforms the pre-training of Language Models as it takes out the RNN and trained the model on the Attention mechanism only. All of these models were unidirectional, but Transformer based Deep Bidirectional Language Model commonly known as BERT from the n-gram Language Model has outperformed the language understanding tasks by using Mask Language Model (MLM) and Next Sentence Prediction[3] and exceed in performance from the previous unidirectional language models and produced SOTA results. It is a cutting-edge approach to natural language processing NLP that has recently achieved state-of-the-art performance on all NLP tasks. BERT was introduced in the year 2018 and by the start of the year 2020, all English queries began to be processed by BERT as it is successfully become the part of the Google search system to understand user queries in a better way [3]. BERT, along with GPT-2, XLNet and other Transformer models, is one of the most interesting advancements in NLP and AI recently.

Another advantage proposed by BERT is that it is a language model which needs to be pre-trained on a large unsupervised dataset only once and then just requires to be fine-tuned on small domain-specific datasets. In the light of the literature, along with other domains like medical, scientific, clinical and business, BERT has also been deployed for performing sentiment analysis on Twitter data by Noureddine Azzouza et al [4] whereas AD Dubey et al. used BERT to perform sentiment analysis specifically on COVID-19 [5]. Apart from the general as well as COVID-19's specific sentiment analysis, literature compels the researchers to perform sentiment analysis specifically on COVID-19 vaccination too. Therefore, to analyze the reaction of the public towards the vaccination, it is imperative to perform sentiment analysis on Twitter tweets related to the COVID-19 vaccination.

1.1 Background & Motivation

The Coivd-19 pandemic has undoubtedly had and will continue to have profound impacts on households and corporations in advanced economies. This pandemic puts a great impact on every industry, whether it is an educational sector, governmental bodies, economy, supply chain, marketing firms, etc. Every individual is suffering from this pandemic, a panic spread among the people, people lost their jobs, people's businesses closed, food shortages occurred, hospitals were filled with patients. But with the passage of time fear and resistance of this pandemic ends. The government takes strict actions to control this pandemic by doing lockdowns, smart lockdowns, providing relief on food, giving aid to needy ones and instantly providing vaccinations. World Health Organization (WHO) continuously releases the latest news regarding the Covid-19 outbreaks. The government of each country has tried to get its people vaccinated as soon as possible. But people have mixed reactions towards Covid-19 vaccinations. Some people have positive reviews and some have negative. We decided to find out the people's mixed reaction towards vaccination, and for this, we need direct interaction with people. But in this pandemic, we can't interact with the public directly to maintain social distance. We decided to find out how many people are satisfied with this vaccination and how many people are dissatisfied. To find out what people think of the covid vaccine, we collected Twitter tweets. To get realistic reviews about vaccination, we decided to fetch Twitter tweets. Twitter tweets are the most reliable resource to get public mixed reactions towards vaccination. In fact, during extreme lockdown days, the public gets information about the lockdown timing or information about sealed areas that have the most Covid-19 patients. Public publicize their opinions related to Covid Vaccination on Twitter. People communicate with each other and express their opinions easily on Twitter in the daily working environment. Twitter's text format is perfect to make datasets. A lot of work is being done these days on Twitter's sentiment analysis. Because Twitter has become a popular local social networking site. When we merged tweets collected from Twitter in one corpus, it becomes very useful information which is also very helpful to understand the social sentiment of people regarding a particular company/brand. We collect Twitter tweets, shape them into a raw dataset and perform Sentiment Analysis on this dataset.

Sentiment Analysis, mainly recognized as Opinion Mining (OM) is an ongoing field of research in natural language processing (NLP) particularly aimed at analyzing subjective and textual information to extract evaluative, judgmental, or behavioral knowledge for the computational treatment of opinions and sentiments of individuals towards a particular subject or element[1]. Sentiment analysis is performed to know people's behavior or sentiment for a particular element. Sentiment analysis is performed on textual data and there are many data sources like Review Sites, Blogs, Forums, G oogle Play Android Application Store, and Micro-Blogging services like Twitter.

Sentiment analysis can be performed through various techniques including Machine Learning (ML), Information Theory and Coding (ITC), Decision Tree (DT), Semantic Orientation (SO), Natural Language Processing (NLP) as well as through Hybrid approaches too. From these mentioned techniques of Sentiment analysis, we select the Natural Language Processing (NLP) technique. NLP is a vast field and has multiple approaches for sentiment analysis, like Bag of Words, Hidden Markov Model (HMM), Part of Speech Tagging (POS), N-gram Algorithms, Large Sentiment Lexicon Acquisition, Parsing Techniques (Top-Down Parsing and Bottom-Up (Parsing) and Bidirectional Encoder Representations from Transformers (BERT). We choose BERT out of all these NLP Techniques, because of its outperformed results on many tasks. BERT is a state-of-the-art open-source framework for NLP tasks. BERT's key novelty of a transformer is the bi-directional ability to train, an attention model to a Language Model. BERT has a more profound perception of language framework and stream than the solo route (uni-directional) model. BERT was introduced in the year 2018 and by the start of the year 2020, all English queries began to be processed by BERT as it is successfully becoming the part of Google search structure to comprehend user queries in a better way.

Previous model based on the traditional RNN and RNN with an attention mechanism. Traditional RNN architecture has a drawback, that the context vector is of a fixed length and cannot store all the relevant information if the input is of large size. And sometimes the danger of information loss. And the Traditional RNN with attention mechanism gives the benefit of calculating the large size of information irrespective of the fixed-length context vector. However, this mechanism is slow, because of its calculation. And it also doesn't give the required performance. Simply if we remove the RNN, because it's taking much time. This architecture improved time complexity and performance by utilizing attention. RNN is too much slow and difficult to train irrespective of the transformer-based models. Transformer is the evolution of the encoder and decoder architecture. In a transformer, the input sequence can be passed in parallel. Insert all the words of the sentence simultaneously and get all the word embedding simultaneously. Normal encoder-decoder uses RNN but Transformer does not use. Transformer achieves faster train time because the Transformer reduces the sentence calculation. Transformer key consideration is a calculation at once by parallelization instead of calculating again and again from beginning to end of the input. Insert all the words of the sentence simultaneously and get all the word embeddings simultaneously.

BERT practice transformer, an attention mechanism works on the learning of contextual context between words in a text. In its raw form, Transformer contains two distinguished mechanisms, an encoder and a decoder. But for BERT we only required an encoder mechanism. Another advantage proposed by BERT is that it is a language model that needs to be pre-trained on enormous unsupervised data only once and then just requires to be fine-tuned on small domain-specific datasets.

Pre-Training of the BERT[6, 7] model on a giant unsupervised dataset is very expensive and demands a lot of pretraining time. Training of BERT for 1M steps on a big unsupervised dataset with a smaller batch size and smaller learning rate required 64 hours of training. BERT pre-trained with longer sentences makes the pretraining very expensive in a computational way. BERT training is done in two phases, pre-train BERT to comprehend language and fine-tune BERT to acquire explicit tasks. While the pre-training model learns what is language and context, and in fine-tuning BERT model learns about, "I know the language and how to resolve the issues.

We present a new modified model replica of BERT with different parameters. Due to the lack of a publicly available BookCorpus dataset, a BERT is pre-trained from scratch only on Wikipedia's (2100M) (13GB) textual data and fine-tuned on the newly developed dataset. To make BERT more effective, a model called BERT_1 is pre-trained on Wikipedia (2100M) as well as on the newly developed unsupervised COVID-19 Vaccination Twitter Tweets (70K) dataset and then further fine-tuned on CV-SAT. Based on the findings of Shah et al.[5], this paper presents a Twitter-Modified BERT (TM-BERT) model to be pre-trained on a smaller Wikipedia (530M) and COVID-19 Vaccination Twitter Tweets (70K) dataset along with the following modifications in the BERT model: A deeper model with a bigger attention head, large batch size with smaller steps and a large vocabulary. The BERT model achieved (0.70) accuracy, the BERT_1 model achieved (0.76), a (6%) improvement, whilst our proposed TM-BERT achieved (0.89) accuracy, a (19%) improvement.

1.2 Problem Statement

To utilize advanced Natural Language Processing based models for the analysis of Covid-19 Vaccination Twitter data for Sentiment Analysis. To introduce a deep bidirectional model that uses limited resources, reduces the pre-training time and generates better outcomes.

1.3 Aims and Objectives

The foremost aims of the research are as follow:

- Developing a new unsupervised COVID-19 Vaccination Twitter Tweets (70K) dataset for pre-training.
- Developing a new COVID-19 Vaccination labeled dataset for fine-tuning.
- BERT model pre-trained from scrape with different hyperparameters.
- Pre-training of BERT model on the task-specific dataset and formerly fine-tuned on the specific downstream task.
- Lessen the size of the model for pre-training, and produce better results.
- Lessen the computational resources required for BERT.
- Making the Pre-Training time short for BERT.

1.4 Structure of the Thesis

This work is organized as follows:

Chapter 2 What is the Language model and how does it work.

Chapter 3 Describe the detailed Literature Review and work of the researcher on Language Models.

Chapter 4 Comprises upon Methodology.

Chapter 5 Comprises upon the Implementation and Results.

Chapter 6 Concludes the thesis and discloses the upcoming scope of this research.

Chapter 2

Language Model

CHAPTER 2: LANGUAGE MODEL

Language modeling is recognized as an emerging field, as it got a huge consideration in the past decade due to enhancement in computer computational powers. Language models are firstly pre-formed, later is calibrated on subsequent tasks by using handful limits instead of studying the entire model from origin. These models are usually pre-trained on unsupervised, semi-supervised, and supervised datasets. Generally, these models have two types i.e., bi-directional, and unidirectional. Language models are constructed on LSTM, CNN, RNN and attention architecture as well.

2.1 Language Model

Language models are described as the essential elements of Natural Language Processing (NLP). These models are the cornerstone of Apple's Siri, Google Assistant and Amazon's Alexa. Language models find out to forecast the likelihood of a series of words particularly in a sentence by utilizing numerous probabilistic and statistical methods. The core objective of these models is to investigate the data by forecasting the words, as these models are considered essential for spell checking, speech recognition and machine translation. If we have an English statement and wish to predict it in Urdu, the language model would assess the entire text before translating each word to Urdu. It's a machine translation element of Natural Language Processing (NLP) that we can't employ with language models.

English: "I like Cricket"
Urdu: "مجھے کرکٹ پسند ہے"

2.2 How Language Models Work

These models are utilized to examine the given words in a sentence and explain the data by changing it to an algorithm, after which it creates the guidelines in the context of natural language. It implements these guidelines to the natural language to produce or forecast sentences. These language models ascertain the key attributes and qualities of the primary language which ultimately assists them to comprehend new sentences.

The probabilistic algorithms models mostly rely on the aim and demand of the model. These models also pertain to the data capacity and the level of mathematics utilization in these models. It can be explained as a model which is developed to examine the probability of search inquiries and outcomes of these inquiries has distant provisions as compared to a model which is utilized automatically.

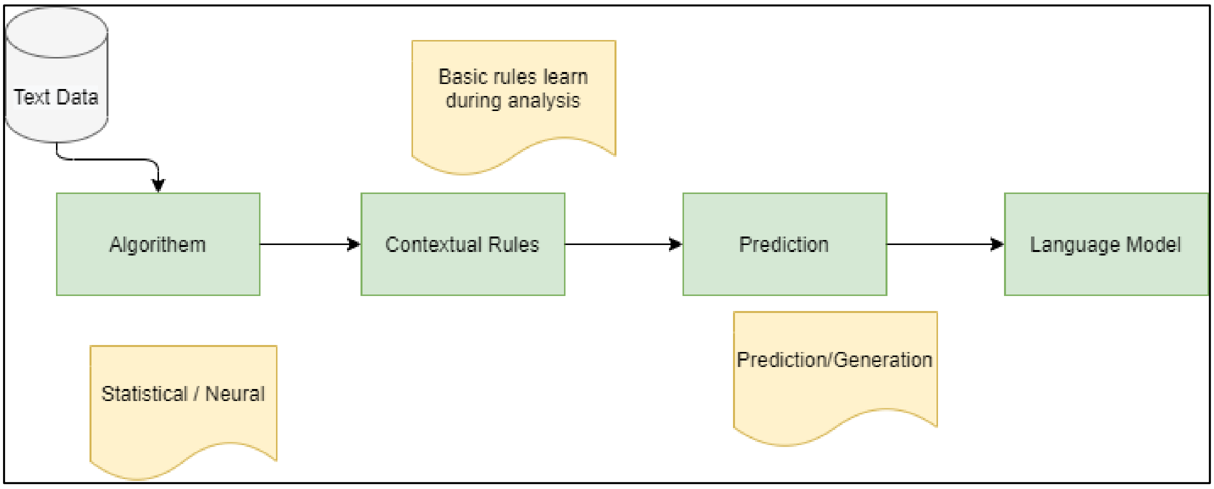


Figure 2 Language Model Illustration

2.3 Types of Language Models

2.3.1 N-gram

N-gram is a probability distribution from the field of computational probability. It is used to analyze the sequence of n observations as a sample from a population. These n observations can be words, phonemes, letters and syllables. It describes the weightage by giving the probability to the series of words. If n is equal to 1, it is known as unigram and if n=3 is described as a trigram. For example, "I love singing" has the n size of 3. It means there are 3 words in the sentence representing n size. This model uses input in a single direction either from right to left or from left to right. This is the prime reason for the expensive training of these models as the n size grows, the memory standards and computational capacities will also grow at the same time. This issue is recognized as the limitation for these models' utilization. It is very difficult in these models to create differences in the sentence when a sentence has the same words but different meanings. For example, the 'bank' word has an independent explanation for "bank account" and "bank of the river". For example, "I am sitting near the canal" will initiate to analyze the input from "I" and ends on "canal".

2.3.2 Unigram

As discussed above, unigram is usually utilized for retrieving information in a simple model where n=1. Unigram avoids conditioning factors as the calculation is very simple and it independently examines each word. Unigram is utilized for sthe probability query model which expresses the probability of query outcomes in a documented form. It takes a single word in the series as n=1. For example, we have a sentence “I love singing” then the unigram will estimate each word independently.

I love Cricket			
Unigram	I	Like	Cricket

2.3.3 Bidirectional

These models understand the input generally from both sides, so they can examine the data from right to left and left to right as well. In this model, n size equals 2 and it evaluates every word included in the series. This model estimates a single word with other words in sentence series. Due to the bi-directionality nature, these models are more accurate as compared to unigram. As we discussed the sentence “I am standing on the bank of the river” this model will understand the input from both sides i.e., backward, and forward like “I” and “river”. It explains here that the bank is expressed for the river, not for the account. The sentence structure is not increased in bidirectional models and this model doesn’t grow with enhancement in sentence size as well as the required capacity remains minimum in contrast to n-gram models. These models can estimate the sequence of words with other words included in the sentence. So, these models are primarily utilized in the machine learning process, for example, Google and YouTube search engine displays bidirectional models to estimate the search queries and results queries

2.3.4 Exponential

These models are also described as maximum entropy models as they depend strongly on entropy. It is developed on the entropy standard which depicts the selection of probability distribution with multiple options and accuracy. The model is considered unique if it is based on zero assumptions and chaos. These models are considered complex as compared to n-gram models where we just input algorithms and run the basic model. These models change the input sentence into an equation after that it evaluates the text by compounding the n-gram model and feature components. This model is unable to explain the parameters which are in unclear pattern as compared to examining individual n-grams and intensifies parameters and attributes for outcomes. Exponential models are designed for maximum entropy which reduces the proportion of statistical choice, and it is developed to enhance the user trust level by offering relative outcomes.

2.4 Pre-training

Model pre-training is similar to human training in which human understands the facts from prior experiences and after that, it implements the expertise to resolve other activities without understanding the scratch. It simply explains as a model is attached and trained with a dataset to generate parameters that are employed to resolve other activities. If there is an activity of classification, we have to train the model according to the requirement, produce parameters accordingly and randomly modify the weights. Due to random weights, the model is optimized, and we can protect it as well after showing minor mistakes. We have an activity and train the model, particularly on image dataset when this model initiates to make a minor mistake and it is showing minor errors then protect it for future utilization.

After saving the model, now implement this similar model to another unique and distant activity regarding image classification. It is no need to train this model from scratch as discussed, we just change some parameters for the new task and the model will be prepared to execute classification on new activity. There is no need to randomly modify the weights as we can utilize the already saved weight normally for this activity, which will ultimately save effort and training time. Generally, pre-training comprises three types such as supervised, unsupervised, and semi-supervised.

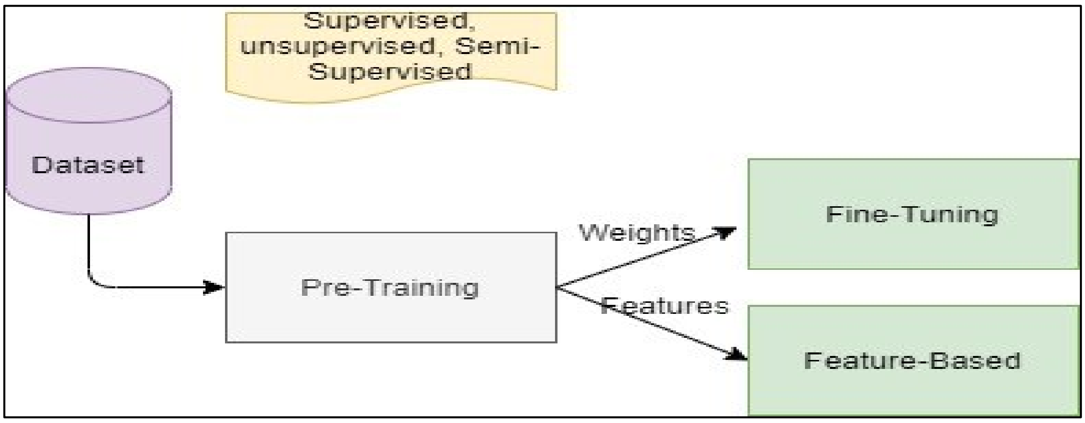


Figure 3 Pre-training Illustration

2.4.1 Supervised Pre-training

It is a learning type where the model is specifically trained and operated on supervised data. These types of datasets have a dependent 'y' value against all independent 'x' values. In short, every value in the dataset contains a target value. These types of models explain accurate results by using labeled data. These datasets also have limitations due to the restricted availability of supervised data because these types of datasets use language models to train huge data. So, it may also execute on other small activities. The shortage in supervised datasets influences the results and then it is used for the small activities where pre-training doesn't cause any issues.

2.4.2 Unsupervised

As discussed, the availability of limited datasets enforces language models, and these models use unsupervised data for pre-training. These data are very large such as books, news, Wikipedia, corpus, etc. This dataset has values of 'x' but no values for 'y' which means every value hasn't a target value. These datasets perform better as compared to supervised datasets because these datasets produce unique parameters through which they can forecast other unsupervised activities.

2.4.3 Semi-supervised

Semi-supervised training involves the benefits of both unsupervised and supervised pre-training as both are pre-trained on unlabeled and labeled datasets. It performs in certain conditions where it is a need to run unlabeled and labeled datasets. These models have attributes of unsupervised and supervised datasets, and they will execute better on different downstream activities. Semi-supervised datasets are used in rare cases as some models are pre-trained on these datasets.

2.5 Fine-tuning

Fine-tuning is described as a model which is pre-trained on utilizing some data after that it gives different weights and produces parameters that are used for later usage. As this model is implemented on different downstream activities, there is no need to train all data from the scratch. Ultimately, it saves training time and effort in the models. The model employs prior experience on the new issues.

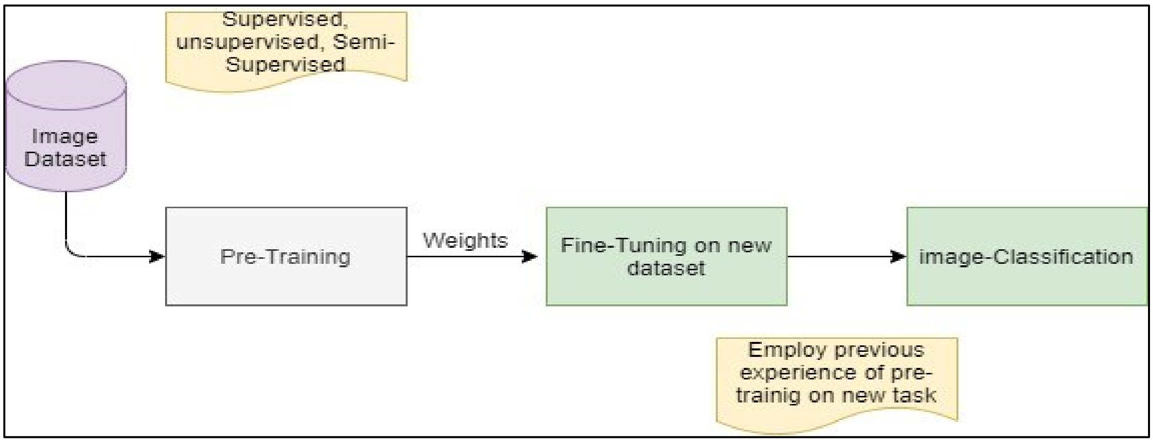

Supervised, unsupervised, Semi-Supervised
Image Dataset
Pre-Training
Weights
Fine-Tuning on new dataset
image-Classification
Employ previous experience of pre-trainig on new task

Figure 4 Fine-tuning Illustration

If a model is pre-trained for image classification, it uses fewer weights to process it, when it is realized the errors are going to minimize, we save the model for analysis in the future. After saving the model, we just modify a few parameters to analyze the model.

2.6 Summary

The language model is described as an area in which language model is firstly pre-formed in a huge quantity of data, later it comes under the fine-tuning for other related activities. These models portray the basic information used during pre-training and transfer this information to downstream activity as the humans perform it. Pre-training language models are used to firstly investigate the text, after that feeds as an algorithm for making vocabulary from these models and then implement these models to understand the language vocabulary like other activities human utilize prior experiences in resolving distant issues known as fine-tuning. For implementing these models on downstream activities, we have made fine-tuning and attribute-based techniques but due to difficulty in the model size and feature component is restricted as mostly language models utilize the fine-tuning technique in downstream activities. Literature reveals three types of pre-training such as unsupervised, semi-supervised and supervised pre-training.

Due to the difficulty and shortage of human-related datasets, semi-supervised data generally contains extensive models and utilizes unsupervised pre-training. Fine-tuning is still used in examining the supervised datasets. There are five types of language models such as n-gram, unigram, bi-directional, exponential and continuous space. N-gram and bi-directional models are recognized as the best models due to their best outcomes and frequent utilization. N-gram expresses the series in one direction either from right to left or left to right. On the other hand, the bi-directional model expresses the inputs from both sides to appropriately investigate the criterion as compared to the unidirectional investigation. Language models are firstly developed on CNN and after the evolution of LSTM and RNN, it is moved towards RNN. It is utilized as an attention instrument with some contextual direction to explain all inputs in a series while all models developed on RNN were found as unidirectional. Overall, the contextual directions of related models were limited in scope as these models haven't the capacity to understand words length and the required memory for models. Transformer architecture eliminates RNN and it is solely utilized as an attention instrument which permits this design to operate bi-directionally. On the other hand, BERT was considered as the vital bidirectional model operating as ELMO but not considered bidirectional. It explains the input from both sides and purely focuses on inputs from right to left and left to right. BERT employed MLM function which confines the model to explain inputs from both sides and secure it to create a multilayer background. It also employs NSP which permits BERT to enhance the background and a strong association among sentences.

Chapter 3

Literature Review

CHAPTER 3: LITERATURE REVIEW

In Natural Language Processing tasks, language modeling has been a fascinating field for researchers. Language Models trained on unlabeled (unsupervised text) and formerly fine-tuned on labeled datasets transform the language modeling[8, 9]. The benefit of using these techniques is that we can easily change the parameters a little and prepare them for the downstream tasks like GLUE, SquAD, etc. There are different training methods, and each method is designed differently with a purpose that comprises machine translation [10, 11], language modeling [8, 9], and masked language modeling (MLM)[6, 12]. Models used distinct approaches for fine-tuning to accomplish downstream tasks[13], some used multi-tasked[14] entity embeddings[15] and different autoregressive pretraining[16, 17]. There are multiple types of pre-training objectives for Language Modeling. In the Auto-Encoding type, the model rebuilds the original data from corrupted inputs. In Auto-Regressive type, the model uses a probability distribution i.e: remembers the previous states and in partially Auto-Regressive model uses only state. In Auto-Encoding and Auto-Regressive type, the model gives the objective in which mortifying, and knowledge of preceding values preserve. In Auto-Encoding and Partially Auto-Regressive type, models present the objective in which corrupting and partially knowledge of previous values preserve. Most of the previous models were uni-directional and can read the Input from only one direction either it's left-to-right or right-to-left. BERT[6] is a forerunner bidirectional language Model. BERT executed the concept of MLM and NSP. Before that BERT ELMo [18] was called a bidirectional model but it was not bidirectional as it combined both right-to-left and left-to-right data that doubles the data and allows the model to see itself redundancy of data and calculative power needed for ELMo was significantly higher than BERT. These language models were based upon Recurrent Neural Network (RNN) architecture. RNN architecture has a drawback, that the context vector is of a fixed length and cannot store all the relevant information, if the input is of large size and sometime the danger of information loss [19].gives the concept of the only use of attention, simply now eliminate the RNN, because it's taking much time. This architecture improved time complexity and performance by utilizing attention. This new architecture can recalls a long sentence.

3.1 BERT Trained with Different Strategy

ALBERT, a lite BERT pre-trained on less computational resources than BERT. ALBERT has 18x fewer parameters than BERT-large and pre-trained 1.7x faster. ALBERT has a maximum input length of 512. The size of the Vocabulary was 30K and using SentencePiece this vocabulary was tokenized. For pre-training purposes, Cloud TPU V3 was used. The number of TPU ranged from 64-1024, depending on model size. ALBERT differentiate from BERT in the following:

In Factorized Embedding Parametrization, the generous vocabulary embedding matrix decomposes into small matrices. Hidden layers size H was separated from the size of vocabulary Wordpiece embeddings E. By doing this, the size of hidden layer size was increased deprived of increasing the parameter size of vocabulary embeddings. Practically, the Vocabulary size V has to be large as required by NLP. And if the Hidden Layers H ties with embedding E, the size of E increased as the size of H increased. Hidden layers are context-dependent and WordPiece embedding is context Independent. By decomposition, the embedding parameter is lessened from Big O of (V*H) to a smaller Big O of (V*E + E*H). Parameter efficiency is improved by sharing cross-layer parameters. This prevents the parameters to increase with the depth of the network. There are multiple ways of sharing, sharing feed-forward network only, Sharing attention parameters only and sharing both. Configuration of the model is no deeper than a 12-layer if cross-layer parameters sharing is done. A significant feature of language comprehension is to sustain inter-sentence modeling. "A loss base mainly on coherence" is purposed. For this ALBERT uses SOP loss, which evades matter forecast and emphases demonstrating inter-sentence consistency. SOP loss practices the same technique as BERT NSP loss, +ve for two sequential sections from the same documents and -ve for the same two sequential sections but with orders transacted. NSP performs not very well on SOP tasks, but SOP out-perform NSP tasks[20].

RoBERta, a robustly enhanced BERT model, pre-trained on the huge dataset. RoBERta was pre-trained on a novel dataset CC-News which is 76 GB. Model pre-trained with dynamic masking and pre-trained full sentences without NSP loss. This model used large-mini batches and Byte-Pair Encoding instead of Wordpiece Encoding[21].

KoreALBERT, a new model is pre-trained and fine-tuned on Korean Language Datasets. KoreALBERT-base has 12M parameters, 12 Layers, hidden layers 768, and 12 attention heads and an embedding size of 128 dimensions. KoreALBERT-large has 18M parameters, 24 layers, hidden layers 1024 and 12 attention heads. This model pre-trained on Web News, Korean

Wikipedia, NamuWiki, and BookCorpus. All these datasets are related to Korean Language. For fine-tuning, KoreALBERT evaluates the following six NLP tasks. KorNLI, KorSTS, Sentiment Analysis (NSMC), Paraphrase Detection (PD), EMRC, and NER. This model introduced a new technique Word Order Prediction (WOP). By this technique, we can check intra-sentence word orderings. To forecast the correct order of scuffled tokens a cross-entropy is used by WOP. Previous techniques MLM and SOP are also used by WOP. An additional linear classifier atop the encoder output is added for WOP. Using softmax, this added layer forecasts the likelihood of the original spot of words in the sentence. Tokenize the corpora into sub-words using SentencePiece tokenizer[22].

BioBERT, a first domain-specific model is proposed, which is pre-trained on a wider range of the biomedical dataset. Pre-training on a medical dataset supports comprehension of the complicated biomedical text. A new WordPiece vocabulary is created using WordPiece tokenization based on biomedical corpora. BioBERT pre-trained on the following text datasets, English Wikipedia has 2.5 B lines, BooksCorpus contains 0.8B lines, PubMed Abstracts contains 4.5B lines and PMC Full-text articles contains 13.5B lines. BioBERT pre-trained for 23 days on biomedical corpora on 8 NVIDIA V100 (32GB) GPUs. Max sentence length 512, Mini batch size 192 resulting in 98,304 words per iteration. The time is taken by the model is10 days to pre-train BioBERT v1.0 on the respective dataset (+PubMed + PMC). And 23 days to pre-train BioBERT v1.1 on this (+PubMed) dataset. BioBERT (+PubMed + PMC) pre-trained for 470K steps, BioBERT v1.0 (+PubMed), pre-trained with PubMed for 200K steps, BioBERT v1.0 (+PMC), pre-trained on PMC for 270K steps and BioBERT v1.1(+PubMed), pre-trained on PubMed for 1M steps. For fine-tuning, a solitary NVIDIA Titan Xp (12GB) GPU is used for the individual task. Different batch size of 10, 16, 32, or 64 was carefully chosen and a learning rate of 5e5, 3e5, or 1e5 was selected. The remaining all hyper-parameters are the same as those used for BERT. BioBERT performs significantly on following biomedical text mining tasks. BioBERT perform fine-tuning on the following tasks and achieved the best results. It attained 0.62% F1 perfection on BioMedical NER, 2.80% F1 perfection on Biomedical Relaxation Extraction (RE), and improved the Biomedical Question Answering task with 12.24 MRR improvement[23].

SCIBERT, a new domain-specific model is proposed. BERT grounded model pre-trained on the enormous scientific dataset. A new WordPiece vocabulary is constructed on a Scientific dataset using SentencePiece library named SCIVOCAB. Developed both cased and uncased vocabulary and the size of the vocabulary is 30k. SCIBERT Model is pre-trained on 1.14M papers from semantic scholars. The percentage of papers are 18% and 82% respectively from the computer science domain and biomedical domain. And corpus size is about 3.17B tokens. ScispaCy used to split sentences. Four different versions of SCIBERT are trained, cased, uncased, BASEVOCAB and SCIVOCAB. Pre-training was done in two phases, first as the maximum length of a sentence is 128 tokens, and for the second one maximum sentence length increased up to 512 tokens. The resource used for the pre-training was a solitary TPU v3 with 8 cores. SCIVOCAB model pre-training took 7 days. The maximum length of 128 tokens took 5 days and similarly maximum length of 512 tokens took 2 days. BASEVOCAB model took 2 days, as it's not trained from scratch. Fine-tuning of a model is done using a batch size of 32 in 2 to 5 eras.

Learning rate of $1e^{-5}$, $2e^{-5}$, $5e^{-5,}$ or $5e^{-6}$ was used with a sideways trilateral agenda. The best setting for the model is 2 or 4 epochs and the learning rate of 2e-5. By training a simple task-specific model on top of frozen BERT embedding. Fine-tuning a larger effect on computer science corpus, SCIBERT achieves an F1 result of +5.59 and BERT-base archives +3.17. SCIBERT outperforms Bert-base on BioMedical and scientific tasks. On Biomedical tasks, SCIBERT achieves SOTA results on BC5CDR, ChemPORT, and EBM-NLP and in the Computer Science domain, SCIBERT attained SOTA results on ACL-ARC and NER part of SciERC[24].

Joshi et al.[25] used MLM at span level which practices a single contiguous segment and recapitulates the designated span. It masks random spans as a substitute for tokens like in BERT and used span borderline depictions to predict the masked span without depending on a single token. This model pre-trains on the same dataset used by BERT-$_{\text{large}}$ and acquires (94.6% & 88.7%) on SQuADv1.1 and SQuADv2.0 correspondingly and 82.2% on GLUE task.

Sun et al.[26] introduced a framework for pre-training that first builds the task progressively and then uses a perpetual multi-task approach to extract semantic, syntactic, and lexical knowledge from these tasks. A batch size of 400K was used for 4k steps same as used by BERT$_{\text{Large}}$ model with a 5^{e-5} learning rate was pre-trained on English Wikipedia, BookCorpus, Reddit and on discovery data. The model outperforms the BERT and XLNet on 16 tasks. It attained 80.6% on the GLUE task.

Liu et al.[21] presented a replicate of the BERT model and trained it for a large batch size with a small step size on five large datasets while removing the NSP. In this paper, changes to BERT hyperparameters are described as these changes can be applied to all BERT models as needed. It attained (88.9%) on GLUE, (88.9/94.6) on SQuADv1.1, and (86.5/89.4) on SQuADv2.0.

Wei et al. [27] proposed an innovative model pre-trained on a giant Chinese corpus with an effective relative positional encoding whole word masking strategy. This model was pre-trained based on $\text{BERT}_{\text{Large}}$ model using a batch size of 5K, a learning rate of 1.84, and maximum input size of 128 for 25K steps.

Radford et al.[28] propose a model pre-trained on sequential, uni-directional, and multi-directional tasks using an explicit self-attention mask and collective transformer network and then finetuned on language understanding and generating tasks. This model was pre-trained based on the $\text{BERT}_{\text{Base}}$ settings while using no Next Sentence Prediction (NSP), batch size of 7680, 6e-^{4} learning rate and 128 maximum input size were used for 0.5M steps. It attained the best results respectively, 87.3% on GLUE, (87.1/93.1) on SQuADv1.1, and (83.3/86.1) on SQuADv2.0.

Wang et al.[29] presented a model pre-trained with a stacking algorithm. It conveys information from shallow to deep models while focusing self-attention on dissimilar layers and different positions that allow it to gain local attention and beginning of sentence distribution. StructBERT model pre-trained based on $\text{BERT}_{\text{Large}}$ model with the batch size 32 and maximum input size 512 for 1M steps. It attained results respectively, 86.7% on GLUE and (87/93) on SQuADv1.1.

Jiao et al.[30] proposed a model where language knowledge is transferred from teacherBERT to studentBERT using the transformer distillation technique. This model has a two-stage learning context in which studentBERT receives inclusive and precise knowledge from teacherBERT by using 28% fewer parameters than teacherBERT. TinyBERT framework was 3 input layers, 312 Hidden Layers and 12 Attention Heads and this model attained (76.5) GLUE, (79.7/87.5) SQuADv1.1, and (69.9/73.4) on SQuADv2.0.

Liu et al.[31] presented a model which used cross-layer sharing and learns from a wide range of activities that allow them to learn general representation through familiarity with new areas and downstream tasks. This model has an advantage from a great quantity of cross-task information and standard presentations that familiarize with innovative techniques and fields. It is based on the setting of BERTLarge with no NSP and 128 maximum input size. It attained 86.4% on the GLUE task.

Clark et al.[32] presented a model that masked the input with a rational substitute through pre-training using a trivial genitor network. This forecasts whether every masked input is substituted by a production sample or not and this makes it four times quicker than BERT. Electra was pre-trained using the setting of the $\text{BERT}_{\text{-Large}}$ on 126GB of data with a batch size of 2K, maximum input size of 512 and 2e-^{4} learning rate for 1.75M steps while using no NSP. It attained the results respectively, (89.5) on GLUE, (89.7/94.7) on SQuADv1.1, and (88.0/90.7) on SQuADv2.0.

Wang et al.[33] presented a compressed model that used in-depth self-attention knowledge followed the distillation process. It is based on the setting of $\text{BERT}_{\text{Base}}$ model, with a batch size of 1K, a learning rate of 5e-^{4} and 33M parameters for 400k steps while using no NSP. A maximum input size of 512 was allowed. 81.7 FI score on SQuADv2.0 was achieved by the model.

Table 1 Overall Result

Paper	Model Title	Corpus Size	Tokens	Dataset Used for Pre-Training	Model Type	Sentence Learning	Cross-layer Parameter Sharing
[20]	ALBERT	13GB	3.8B	Books corpus + Wikipedia	Auto Encoding	SOP	True
[21]	ROBERTa	160 GB	2.2T	BOOKCORPUS+WIKIPEDIA+CC-NEWS+OPENWEBTEXT+STORIES	Auto Encoding	None	False
[22]	KoreALBERT	43GB	4.4B	Web News, Korean Wikipedia, NamuWiki, BookCorpus	Auto Encoding	WOP	True
[23]	BioBERT	29.9 GB	21.3	Wikipedia, BooksCorpus PubMed Abstracts PMC articles	Auto Encoding	NSP	False
[24]	SCIBERT	10GB	3.17B	SciCite, BC5CDR	Auto Encoding	NSP	False

[25]	SpanBERT	13GB	3.8B	Books corpus + Wikipedia	Auto Encoding	None	False
[34]	MT-DNN$_{kd}$	13GB	3.8B	Books corpus + Wikipedia	Auto Encoding	None	True
[30]	TINYBERT	13GB	3.8B	Books corpus + Wikipedia	Auto Encoding	NSP	False
[28]	Nezha		10.5B	Chinese Wikipedia+ Baidu Baike+ Chinese News	Autoencoding+ Auto regressive	NSP	True
[31]	MT-DNN	13GB	3.8B	Books corpus + Wikipedia	Auto Encoding	NSP	True
[33]	MINILM[a]	13GB	3.8B	Books corpus + Wikipedia	Auto Encoding	None	False
[17]	XLNet	126GB	32.89B	Bookscorpus+Wikipedia+Giga5+ Clue Web 2012-B+ Common Crawl	Autoencoding+ Auto regressive	None	True
[35]	T5	29TB	34B	Colossal Clean Crawled Corpus	Auto Encoding	None	True

[36]	FreelbROBERTa	160GB	-2.2T	BOOKCORPUS+WIKIPEDIA+CC-NEWS+OPENWEBTEXT+STORIES	Auto Encoding	None	False
[37]	AdaBERT	13GB	3.8B	Books corpus + Wikipedia	Auto Encoding	None	True

Table 2 Overall Hyperparameters

Paper	Batch Size	Max Sequence	Learning Rate	Step Size	Parameters	Layers	Hidden	Attention Head
[20]	4096	512	0.00176	125K	233M	12	4096	128
[21]	2K	512	$1e^{-6}$	125K	360M	24	1024	16
[31]	256	128	$1e^{-4}$	2.4M	340M	24	1024	16
[22]	2048	64 / 128	$1e^{-3}$	125K	12M	24	768	12
[23]	256	512	$5e^{-5}$, $3e^{-5}$ or $1e^{-5}$	1M	21.3B	12	768	12
[24]	32	512	$2e^{-5}$		3.17B	12	768	12
[17]	2048	512	$1e^{-5}$	500K	340M	24	1024	16
[30]	256	128	1	1M	14.5M	4	312	12
[31]	256	128	$1e^{-4}$	1M	340M	24	1024	16
[35]	2048	128	0.01	2.1M	11B	12	768	12
[36]	8K	512	$1e^{-6}$	500K	360M	24	1024	16
[33]	1024	512	$5e^{-4}$	400k	33M	12	768	12
[37]	128	512	$3e^{-4}$	50K	9.5M	24	1024	16
[28]	5120	128	$1.8e^{-4}$	25K	340M	24	1024	16

3.2 Related Work

Twitter has been a valuable source of information nowadays due to its millions of users from all over the globe. Twitter has a large amount of data, and it's unable to classify, filter, read, understand, summarize, and perform any other action manually. It's also a challenging task to solve with Machine learning and Natural Language Processing tools.

Covid-twitter BERT (CT-BERT) model has been developed to perform analysis on the content of Twitter regarding COVID-19. CT-BERT model is founded on BERT-LARGE English Model. BERT-Large is pre-trained on Wikipedia and a free book corpus. To improve performance on subdomains, this model is pre-trained on the dataset of 160M tweets about the COVID-19 collected through the Crowd breaks platform. The corpus was cleaned to remove the tags, usernames, and Unicode emoticons. Duplicate words, duplicate lines, and empty lines are removed. So, the final corpus results in 22.5M tweets that contain 0.6B words. To assess the model's performance on downstream tasks is evaluated on five different datasets, containing Twitter tweets. These datasets are named COVID-19 Category (CC), Vaccine Sentiment (VS), Maternal Vaccine Stance (MVS), Twitter Sentiment SemEval (SE), and Stanford Sentiment Treebank 2 (SST-2). To assess/measure the performance, the F1 score was compared. The final averaged result on five datasets is a mean F1 score of 83%. This shows there is more space for efficiency improvement on the downstream tasks[38].

The Italian BERT XXL, which was pre-trained on plain text large corpus of Wikipedia dump, OPUS corpora, and an Italian part of OSCAR corpus, with a total size of 81 GB containing the 13B words. And finetuning was performed on the notorious dataset SENTIPOLC 2016. This research was divided into a two-stage series, the first stage includes the pre-processing of Twitter dialect into plain text and the second stage manipulates the variants of BERT to finetune and classify the tweets concerning their polarity. In this work, comprehensive sentiment is examined, therefore both positive and negative sentiment analysis is also covered with the overall sentiment of the tweets. The final F1 recorded score was 74.0% for positive and 76.0% for negative and a final averaged a score of 75.0%[39].

Adversarial training of Bert named BAT is performed to utilize the adversarial training to attain the finest result in sentiment analysis. This paper outperformed on two tasks of Aspect Based Sentiment Analysis, Aspect Extraction, and Aspect Sentiment Classification. We keep the record of only Aspect Sentiment Classification. Dataset used for the experimental purpose was SemEval 2014 task 4 [32] and SemEval 2016 task 5. This experiment was performed on GPU

(GeForce RTX 2070). The final maximum accuracy result produced by the BAT model was 82.27% with BERT-BASE initialization and 85.6% with BERT-PT initialization[40].

BERT outperformed on many tasks with minimum changes in hyperparameters or by pre-training it on the different task-based datasets. Hu Xu et al, experimented on a new task, exploiting the client reviews into a bulky source of knowledge and named it as Review Reading Comprehension (RRC). Created a new dataset ReviewRC. Enhanced result of Aspect Extraction (AE), Aspect Sentiment Classification (ASC) and outperformed on RRC, but we consider the result of ASC only. Post-training was performed using the following parameters, maximum length set to 320 with batch size equal to 16, and Adam Optimizer's learning rate 3e-5. The training was performed on GPU. The maximum final accuracy given by BERT-PT was 84.95%[41].

A monolingual BERT model was developed including the Dutch language named as BERTje. Pre-training was performed on famous corpora having high-quality Dutch text. Books, TwNC, SoNaR-500, Web news, and Wikipedia. BERTje use SOP instead of NSP. And for MLM, BERTje masked consecutive words instead of masking a single word randomly. BERTje outperforms fine-tuning on different tasks including NER, POS tags, and classification tasks. From this paper, we only take into account the classification results of sentiment analysis. The accuracy scores calculated on 110k Dutch Book Review Dataset was 93.0%. This result was very much close to ULMFit model[42].

BERT model was pre-trained on two social media platform texts, including tweets and forum texts. All the same, hyperparameters were used except NSP, which was used by the original BERT model. Two pre-trained models were introduced, TwitterBERT and ForumBERT, trained on twitter datasets and forum text on business review respectively. Improved final results were given by both pre-trained models as compared to the original BERT. This shows the importance of selecting in-domain source datasets. TwitterBERT achieves an F1 score that was 80.8% on Twitter datasets and ForumBERT achieves a 93.4% F1 score[43].

Sentiment analysis was performed on covid19 tweets, tweets taken from all over the world, and specifically tweets related to the Indian region. Dataset was divided into two portions, then labeled. Dataset is portioned into training and testing sets. Dataset was pre-trained by both BERT-base and BERT-large model and then finetuned it. The validation accuracy (MCC Validation Accuracy) score was 93.8 %[44].

T-BERT Model was present to perform sentiment analysis of Micro-blogs Integrating Topic Model. Dataset was extracted from Twitter that's around 40k raw microblog. The data was further pre-processed. An experiment was performed on BERT-base with the same parameter used by the original BERT model. Furthermore, widely numerical experiments were conducted by choosing different hyper-parameters. The final accuracy was measured separately as 0.96, 0.78, and 0.69 for positive, negative, and neutral data respectively [45].

Covid19 related tweets are extracted from Twitter. Classification for sentiment analysis was performed on two BERT models. BERT-base-uncased achieved a F1 score of 0.70 and COVID-Twitter-BERT achieved F1 score of 0.76. COVID-Twitter-BERT classifier was deployed on more than 85 million tweets. This model was trained to detect self-reporting potential cases from Twitter tweets that were not reported to officials in the United States. This model detects 13,714 potential cases [46]

BERTweet, a large-scale pre-trained is presented, which is trained on 80GB corpus containing 850M English tweets and 5M tweets related to COVID-19. BERTweet model outperforms RoBERTa-$_{base}$ and XLM-R$_{base}$. BERTweet outperforms on the following downstream Tweet NLP tasks, POS Tagging, NER, and text classification. The model used the same parameters as the BERT-base and pre-trained based on the pre-training procedure of the RoBERTa for more robust execution. Tokenization of the tweets is done using TweetTokenizer from NLTK toolkit and an emoji package is used to convert emotion icons into text. Normalizations of tweets are executed to remove all the tags, usernames, retweets, and Unicode emoticons. After the cleaning process of the datasets, 845M tokens are left behind. 3-class sentiment analysis dataset from (SemEval2017 Task 4A) and the 2-class irony detection dataset from (SemEval2018 Task 3A) was used for a text classification task. For fine-tuning tasks, AdamW with a learning rate of 1.e-5 is used and a batch size of 32. BERTweet final score was compared with RoBERTA and XLM-R, which is higher than both models. BERTweet final F1 score reported for SemEval2017 Task 4A was 72.8 and 73.8 for hard and soft normalization respectively and accuracy was 71.7 and 72.0. BERTweet final F1 score reported for SemEval2018 Task 3A was 74.6 and 74.3 for hard and soft normalization respectively and accuracy was 78.2 and 78.2 [47].

On two different Twitter datasets, the BERT model is pre-trained and fine-tuned to achieve the maximum possible results. For the English language, SemEval2017 is used and for the Italian language, the SENTIPOLC 2016 (SENTIment POLarity Classification) datasets are used. At first, the novel pre-processing technique for the initialization of the Twitter text is performed to remove all the unnecessary words. The aim is to achieve tweets that are less noisy, clean, and exploit the hidden information which is previously ignored while pre-processing. BERT was pre-trained on Book-Corpus and English Wikipedia for English language and OPUS corpus and OSCAR corpus (only Italian part) for the Italian Language. Then fine-tuned to achieve maximum possible results. Some different change parameter was used, like learning-rate 3e-5, train_batch_size 8, num of train epochs 5, and gradient_accumulation_steps 16. The final accuracy results measured on both datasets were 68% for English tweets and 75% for Italian tweets[48].

TwilBERT, a model is introduced which is pre-trained in Spanish language and Twitter domain. In this model, the BERT model is pre-trained from scratch, and to acquire inter-sentence coherence in the Twitter conversation a Reply Order Prediction (ROP) Signal is introduced. Two different TwilBERT models were presented as TwilBERT-Base (TW-Base) and TwilBERT-large (TW-Large) with different transformer layers and attention heads. TW-Base has 6 Transformer layers and 6 attention heads, while TW-Large has 12 Transformer layers and 12 attention heads. TwilBERT performs comprehensive execution on 14 different classification tasks. The dataset used for the sentiment analysis task was contributed to the 2019 Workshop on Semantic Analysis at SEPLN (TASS). The organizer gives five corpora in different versions of the Spanish language (Spain, Mexico, Uruguay, Perux, and Costa Rica). The final accuracy measured for the TW-base model was 59.14 for the Spain variant, 57.20 for the Costa Rica variant, 63.0 for the Uruguay variant, 48.22 for the Peru variant, and 62.73 for the Mexico variant. And accuracy for the TW-large model was 59.50 for the Spain variant, 59.52 for the Costa Rica variant, 62.88 for the Uruguay variant, 44.06 for the Peru variant, and 63.67 for the Mexico variant. TwiBERT gives the best result for all variants except the Peru variant [49].

TwitterBERT, a four-phase setup with various deviations of the classification model is proposed. This model is pre-trained additionally one step ahead of the original BERT model by arranging strategically a collection of classification models on the edge of the language model's final layer. Intermix BERT word embedding with different variants of word embedding like GloVe, Word2Vec, and FastText. The Four-phase setup of TwitterBERT starts with pre-processing of the datasets by removing the redundant words, emojis, emoticon, URL and hashtags, etc. Pre-training of TwitterBERT is carried out on two datasets, TSA and Sentiment 140. After pre-training on these two datasets, the BERT model is then pre-trained again on 60k tweets taken from each test case datasets. For fine-tuning of the model on down-stream tasks variety of datasets are used (MTSA, SemEval 2013, SemEval 2014, SemEval 2015 , SemEval 2016 , SemEval 2017). The final ensembled F1 score of all the four pre-trained models on these six datasets are 69.26 for MTSA, 72.61 for SemEval 2013, 72.48 for SemEval 2014, 68.23 for SemEval 2015,68.10 for SemEval 2016 , and 71.82 for SemEval 2017. Models give the best result for SemEval 2013[4].

Table 3 Overall Results

Paper	Model	Results
[38]	CT-BERT	83%
[50]	Italian BERT XXL	76%
[40]	Adversarial training of Bert	85.6%
[40]	BERT-PT	84.9%
[42]	BERTje	93%
[43]	TwitterBERT ForumBERT	80.8% 93.4%
[44]	BERT-large	93.8%
[45]	T-BERT	96%, 78%, 69%
[31]	COVID-Twitter-BERT	76%
[47]	BERTweet	78%
[48]	BERT model	68% and 75%
[49]	TwilBERT	64%
[4]	TwitterBERT	72%

3.3 Literature Summary

In Language Modeling, BERT has outperformed many Natural Language Processing tasks. BERT is a transformer-based model, that relies only on the encoder understanding the context of the sentence more precisely than any other model. BERT is pre-trained by using traditional Left-To-Right (LTR) and Left-To-Right (RTL) techniques. Previously ELMO was pre-trained on the traditional way of LTR and RTL separately and then concatenate the results. This results in the ambiguity of the data. BERT uses two objectives which are Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). BERT was pre-trained on the Twitter tweets, as Twitter has a large amount of data, and it's unable to classify, filter, read, understand, summarize, and perform any other action manually. Sentiment analysis was performed on the Twitter tweets using the BERT model. A Covid-twitter BERT (CT-BERT) model was presented to perform an analysis on the content of Twitter regarding COVID-19. This model was pre-trained on the dataset of 160M tweets about the COVID-19 collected through the Crowd breaks platform. CT-BERT outperformed on the domain-specific task of Covid-19 tweets.All the models are grounded on the BERT model but with different changes. Some used both MLM and NSP, some of them removed NSP. Some used domain-specific data for the training of the model and some used general data. Some used the BERT-base setting or some used the BERT-large setting. Some models pre-trained the BERT model for large step sizes or some of them trained for small step sizes. In short, each model used a different setting to optimize the BERT performance. All of above mentioned models were either pre-train on COVID-19 tweets are fine-tuned but none of the model was pre-trained on COVID-19 Vaccination tweets which is the need of time.

3.4 Research Gaps

- Absence of domain-specific downstream task for COVID-19 Vaccination Sentiment Analysis.
- Pre-training of BERT-like models on General and domain-specific datasets can increase performance on downstream tasks.
- Reduction of the general dataset for pre-training may not affect the result (Why not try?).
- Pre-train BERT-like models with attention in a large sample (increasing batch size with attention heads) can improve the performance along with less resource consumption.
- A model pre-train for small steps but along with a large sample of data in one bucket can reduce the Pre-training time.
- A model pre-train with a large vocabulary size, the larger the vocabulary the more words we have. This can help in getting more context for the words.
- A model specifically pre-train on small general and domain-specific datasets with hyper-parameters can improve downstream performance along with less pre-training time, less resource consumption.

Chapter 4

Methodology

CHAPTER 4: METHODOLOGY

In this chapter, we will define the working of innovative BERT, raw collection of datasets and working of our model TM-BERT.

4.1 Working of BERT

BERT is the primary fine-tuned constructed model that attains futuristic routine on the big scale of sentence-level and token-level errands. BERT's key originality is the bidirectional training of a transformer, an attention model to a language model. BERT has a profound acuity of language framework and flows than a single direction model. BERT's key features are as follows, Bi-directional, generalizable, high performance, and universality.

BERT is built upon the transformer architecture, an attention mechanism that works on the learning of contextual relations between arguments in a script[19]. In raw format, a transformer contains two distinguished mechanisms, a decoder and an encoder, whereas for BERT, only an encoder mechanism is required. Unlike other existing language models, BERT is not pre-trained by using the traditional Left-To-Right (LTR) or Right-To-Left (RTL) approach, but rather pre-trained bidirectional. However, BERT uses two objectives which are Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). MLM masks arbitrary symbols and then predicts those tokens during pre-training, whereas NSP predicts whether the next sentence is a continuation of the previous one or not.

4.2 Pre-Training

BERT is not pre-trained via outdated LTR and RTL language models. Though, BERT pre-train by dual unsubstantiated errands concurrently. These are Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). Using the above techniques concurrently, BERT attains a respectable considerate of the language. The MLM arbitrarily masked slightly of the tokens after certain input. For MLM, BERT takings verdicts with arbitrary words occupied masks. The main achievement is the output of these masked tokens. It's like a kind of fil in the blanks, It helps BERTS comprehends a bidirectional framework inside a verdict. To pre-train a profound bidirectional model we arbitrarily mask 15% of all WordPiece mask tokens in each sequence. 80% of which masked by one of the masked tokens, 10% of which random token and 10% of remain unchanged. To train a model which comprehends the verdict affiliation, BERT pre-train for binarized NSP. For NSP, BERT grabs two sentences and it governs whether the next sentence tracks the flow of the first sentence. In the case of two sentences A & B, to determine whether B tracks the flow of A or not. 50% of these sentences tracks the flow of A and it is categorized as (IsNext). And 50% of these sentences do not track the flow of A and it's a random sentence. And it is labeled as (NotNext).

For the involvement of pre-training, we generate a word token, and its input is built by summating up the token, segment and position embedding. Word token inputs are the words of the input text and BERT passes this word token to the embedding layer through token embedding so that the token can be transformed into a vector. Token, Segment and Position Embeddings generate the initial embeddings.

4.2.1 Token Embeddings

The mere purpose of token embedding is to transform a word token into a vector depiction of a static measurement. Firstly, the contribution manuscript is first tokenized earlier while sending it to the token embedding layer. And additional tokens are used before the start of the sentence and at the end of the tokenized lines. For example; " I love RoosterBliss" "[CLS]" , "I" , "love" , "RoosterBliss" , "[SEP]". We add these tokens to explain the input depiction for classification tasks and the parting of input sentences correspondingly. WordPiece tokenization method is used to generates words for the generation of BERT vocabulary of 30, 522 Words.

4.2.2 Segment Embeddings

Segment embedding is the binary number as 1, 0 that is prearranged hooked on a vector. So just to distinguish the inputs in a given pair Segment Embedding is used. For example.

" I love RoosterBliss" "A frozen food chain "

Concatenation and Tokenization

[CLS] I love RoosterBliss [SEP] A frozen food chain

Now label to distinguish

[CLS]	I	love	RoosterBliss	[SEP]	A	frozen	food	chain
0	0	0	0	0	1	1	1	1

Segment Embedding layer has only two representations (0,1). The first vector is 0, and this assigned to all the word of the first sentence, and 1 is assigned to all the words of sentences b.

4.2.3 Position Embeddings

Position embedding is described as the location of an expression inside the verdict that is prearranged into a vector.

By addition of all these vectors collectively, and embedding vector is generated which is further used as the BERT's input vector.

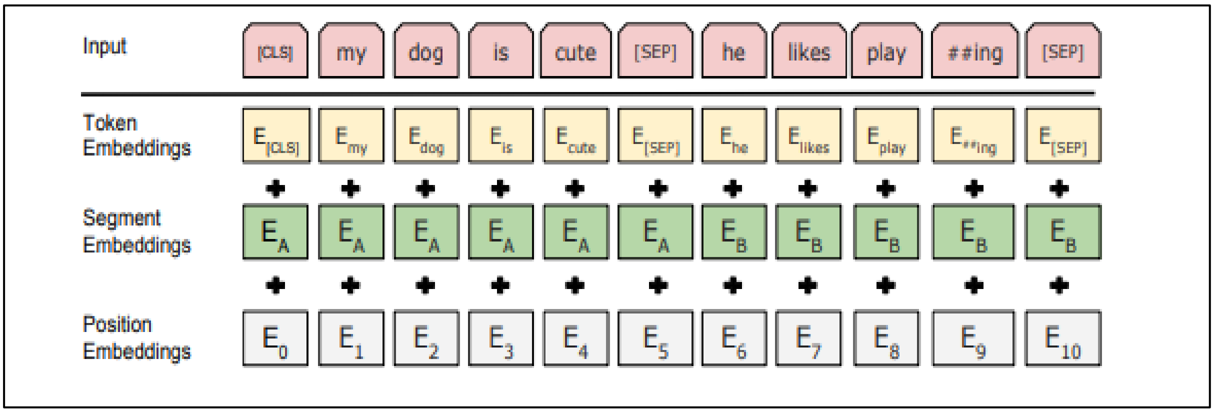


Figure 5 Input Vector

4.3 Fine-Tuning

For domain-specific chore, lump in the activity-specific input and gives its output to the BERT in order to fine-tune entirely the parameters. Model is fine-tuned by adapting the inputs and output layer differently. We perform supervised training dependent on the job we need to resolve. Let's take Question Answering (Fine Tuned Q & A).

4.4 COVID-19 Vaccination Sentiment Analysis Task (CV-SAT)

To fine-tune the BERT model, a new COVID-19 Vaccination Sentiment Analysis Task (CV-SAT) is developed. A Twitter API used 'COVID-19 Vaccination' and 'COVID vaccination' keywords to gather (50K) COVID-19 vaccination-related Twitter tweets shared between the period January 2021 till November 2021. Twitter-preprocessor library of python is used to remove URLs, mentions, emojis, reserve words and smileys, whereas Natural Language Toolkit (NLTK) is used to deal with tokenization. After preprocessing of the dataset, Azure Machine Learning Extension is used to perform labeling on the dataset which is then checked manually. The dataset is labeled as positive and negative where the positive labels are converted to (1+++$+++) and negative labels to (0+++$+++) to ensure its compatibility with BERT architecture shown in Figure 1. As our data is in plain text and in one column, so just to split the data, we used (1+++$+++) and (0+++$+++) labels. If we use 1 or 0, it'll be mixed with the data. The dataset is divided into three different files; Dev, Get-Test and Train file. Below is the sample picture of the dataset.

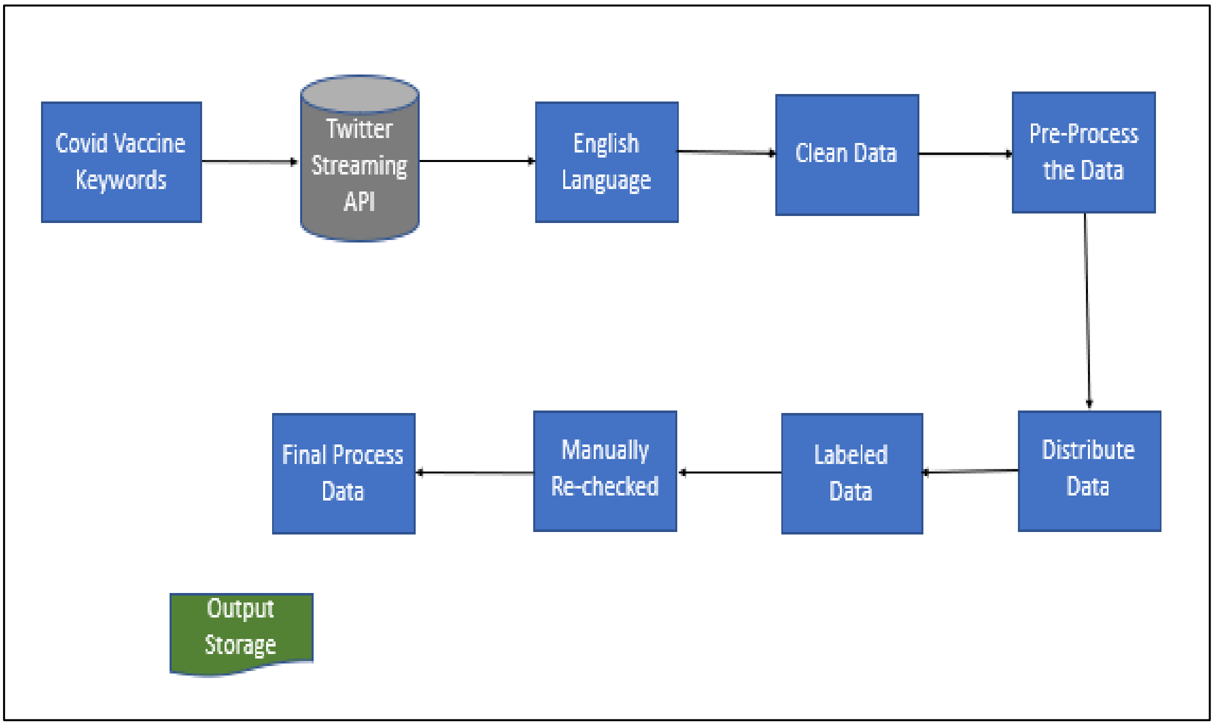
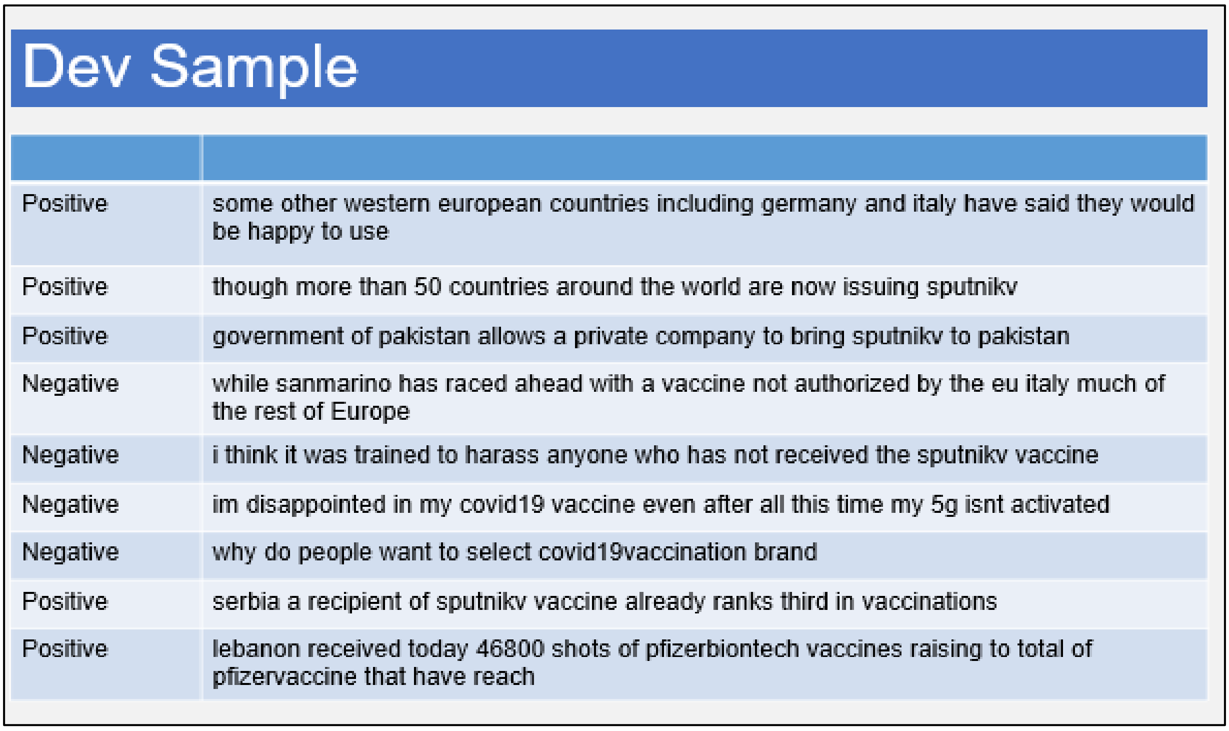


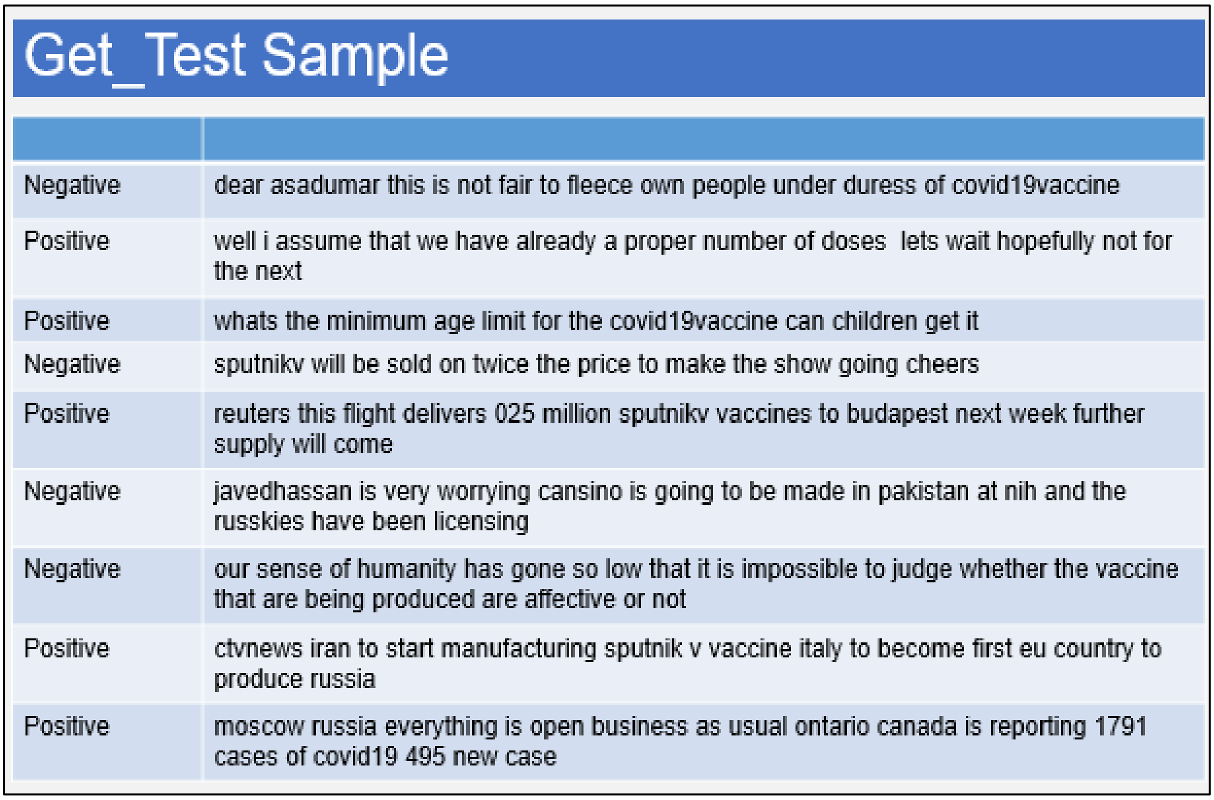
Figure 6 Flow Diagram of Data Collection



Dev Sample

Positive	some other western european countries including germany and italy have said they would be happy to use
Positive	though more than 50 countries around the world are now issuing sputnikv
Positive	government of pakistan allows a private company to bring sputnikv to pakistan
Negative	while sanmarino has raced ahead with a vaccine not authorized by the eu italy much of the rest of Europe
Negative	i think it was trained to harass anyone who has not received the sputnikv vaccine
Negative	im disappointed in my covid19 vaccine even after all this time my 5g isnt activated
Negative	why do people want to select covid19vaccination brand
Positive	serbia a recipient of sputnikv vaccine already ranks third in vaccinations
Positive	lebanon received today 46800 shots of pfizerbiontech vaccines raising to total of pfizervaccine that have reach

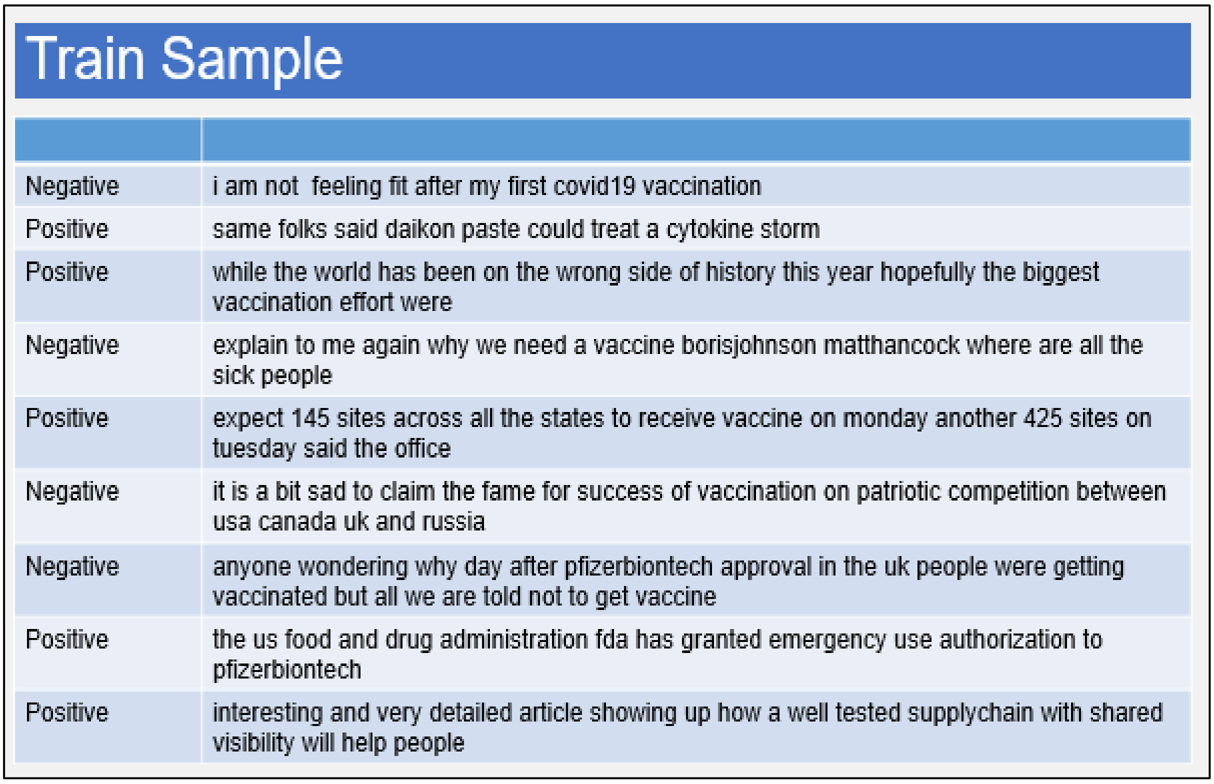
Figure 7 Dev Sample



Get_Test Sample

Negative	dear asadumar this is not fair to fleece own people under duress of covid19vaccine
Positive	well i assume that we have already a proper number of doses lets wait hopefully not for the next
Positive	whats the minimum age limit for the covid19vaccine can children get it
Negative	sputnikv will be sold on twice the price to make the show going cheers
Positive	reuters this flight delivers 025 million sputnikv vaccines to budapest next week further supply will come
Negative	javedhassan is very worrying cansino is going to be made in pakistan at nih and the russkies have been licensing
Negative	our sense of humanity has gone so low that it is impossible to judge whether the vaccine that are being produced are affective or not
Positive	ctvnews iran to start manufacturing sputnik v vaccine italy to become first eu country to produce russia
Positive	moscow russia everything is open business as usual ontario canada is reporting 1791 cases of covid19 495 new case

Figure 8 Get_Test Sample



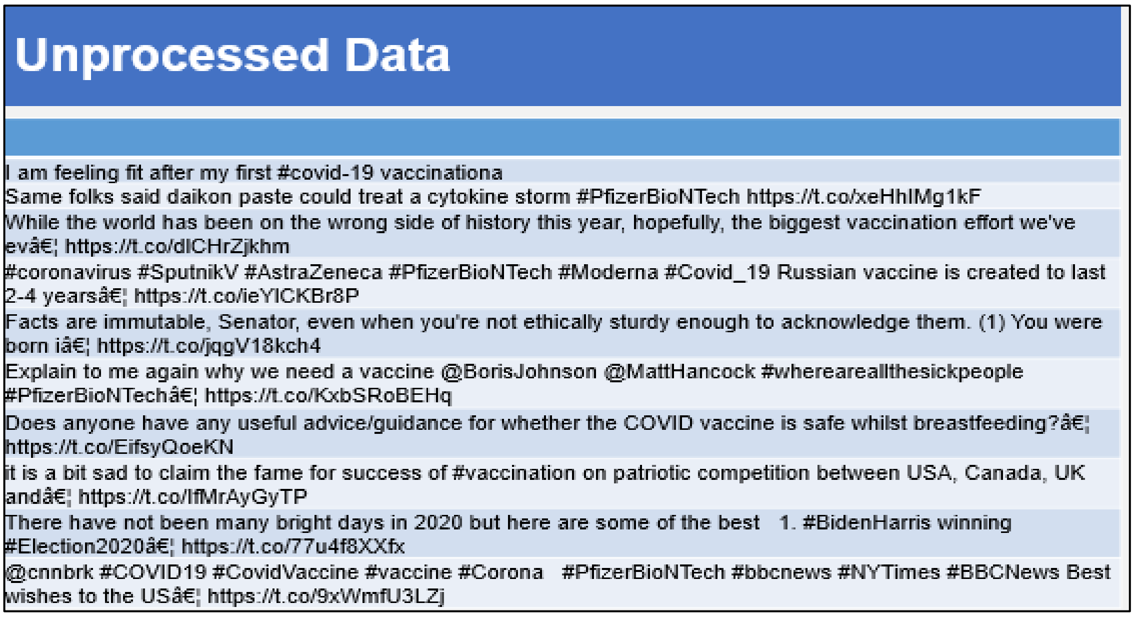
Train Sample

Negative	i am not feeling fit after my first covid19 vaccination
Positive	same folks said daikon paste could treat a cytokine storm
Positive	while the world has been on the wrong side of history this year hopefully the biggest vaccination effort were
Negative	explain to me again why we need a vaccine borisjohnson matthancock where are all the sick people
Positive	expect 145 sites across all the states to receive vaccine on monday another 425 sites on tuesday said the office
Negative	it is a bit sad to claim the fame for success of vaccination on patriotic competition between usa canada uk and russia
Negative	anyone wondering why day after pfizerbiontech approval in the uk people were getting vaccinated but all we are told not to get vaccine
Positive	the us food and drug administration fda has granted emergency use authorization to pfizerbiontech
Positive	interesting and very detailed article showing up how a well tested supplychain with shared visibility will help people

Figure 9 Train Sample

4.5 Twitter Unsupervised Dataset

The Wikipedia dataset is an open-source dataset and publicly available. The unprocessed tweets were collected from Twitter using its API, which resulted in a bustling and vague dataset, due to users' arbitrary, creative, artistic use of social media. Tweets have some specific tokens, i.e., emails, URLs, contact numbers, mathematical numbers, pricing, different time formats, emojis, emoticons, hashtag-words, and special characters like mentions. We pre-process the data and generate the tweets into meaningful sentences and formal form. Twitter tweets (70K) about COVID19-Vaccination are gathered by using an API that contains some specific tokens, i.e., emails, URLs, contact numbers, mathematical numbers, pricing, different time formats, emojis, emoticons, hashtag-words, and special characters like mentions. The data is then pre-processed, and the tweets are generated into meaningful sentences and formal forms. The same procedure, which was performed for the creation of CV-SAT, is followed apart from omitting the labeling task and performing a frequency distribution of words in the data to check the most occurring words along with creating a word cloud too.



Unprocessed Data

I am feeling fit after my first #covid-19 vaccinationa

Same folks said daikon paste could treat a cytokine storm #PfizerBioNTech https://t.co/xeHhIMg1kF

While the world has been on the wrong side of history this year, hopefully, the biggest vaccination effort we've evâ€¦ https://t.co/dlCHrZjkhm

#coronavirus #SputnikV #AstraZeneca #PfizerBioNTech #Moderna #Covid_19 Russian vaccine is created to last 2-4 yearsâ€¦ https://t.co/ieYICKBr8P

Facts are immutable, Senator, even when you're not ethically sturdy enough to acknowledge them. (1) You were born iâ€¦ https://t.co/jqgV18kch4

Explain to me again why we need a vaccine @BorisJohnson @MattHancock #whereareallthesickpeople #PfizerBioNTechâ€¦ https://t.co/KxbSRoBEHq

Does anyone have any useful advice/guidance for whether the COVID vaccine is safe whilst breastfeeding?â€¦ https://t.co/EifsyQoeKN

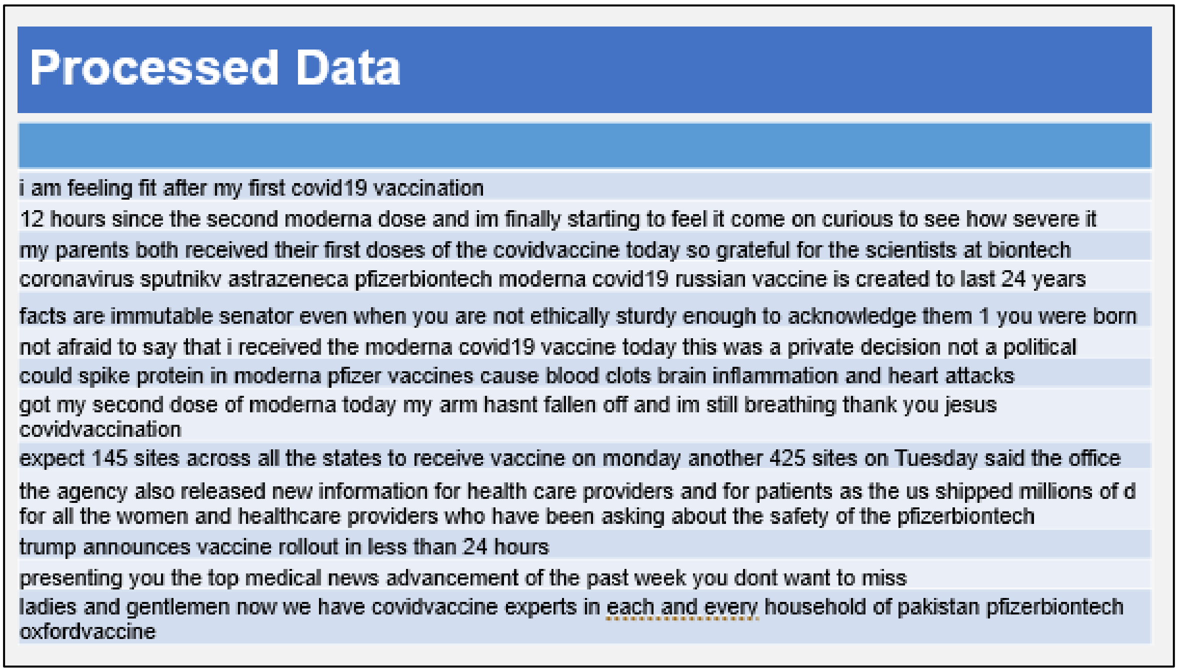
it is a bit sad to claim the fame for success of #vaccination on patriotic competition between USA, Canada, UK andâ€¦ https://t.co/IfMrAyGyTP

There have not been many bright days in 2020 but here are some of the best 1. #BidenHarris winning #Election2020â€¦ https://t.co/77u4f8XXfx

@cnnbrk #COVID19 #CovidVaccine #vaccine #Corona #PfizerBioNTech #bbcnews #NYTimes #BBCNews Best wishes to the USâ€¦ https://t.co/9xWmfU3LZj

Figure 10 Unprocessed Data Sample

Firstly, we gather the tweets using Twitter API. The data is unprocessed and is of no use. We cleaned the data using python's basic library tweet-preprocessor. This library deals with URLs, mentions, reserved words, emojis, and smileys. After cleaning the data, we further pre-process the data using Natural Language Toolkit (NLTK) [51]. This library deals with tokenization, removal of digits, stop words, and punctuation. After all the pre-processing we execute a frequency distribution of words in the data to check the most occurring words and create a word cloud. We fetch almost 125k tweets, but after cleaning and pre-processing only 100K tweets are left behind.



Processed Data

i am feeling fit after my first covid19 vaccination

12 hours since the second moderna dose and im finally starting to feel it come on curious to see how severe it

my parents both received their first doses of the covidvaccine today so grateful for the scientists at biontech

coronavirus sputnikv astrazeneca pfizerbiontech moderna covid19 russian vaccine is created to last 24 years

facts are immutable senator even when you are not ethically sturdy enough to acknowledge them 1 you were born

not afraid to say that i received the moderna covid19 vaccine today this was a private decision not a political

could spike protein in moderna pfizer vaccines cause blood clots brain inflammation and heart attacks

got my second dose of moderna today my arm hasnt fallen off and im still breathing thank you jesus covidvaccination

expect 145 sites across all the states to receive vaccine on monday another 425 sites on Tuesday said the office

the agency also released new information for health care providers and for patients as the us shipped millions of d
for all the women and healthcare providers who have been asking about the safety of the pfizerbiontech

trump announces vaccine rollout in less than 24 hours

presenting you the top medical news advancement of the past week you dont want to miss

ladies and gentlemen now we have covidvaccine experts in each and every household of pakistan pfizerbiontech oxfordvaccine

Figure 11 Processed Data Sample

Below is the Word cloud [52] picture of the Covid-19 Vaccination dataset, this shows the occurrence of most repeated words in the data.

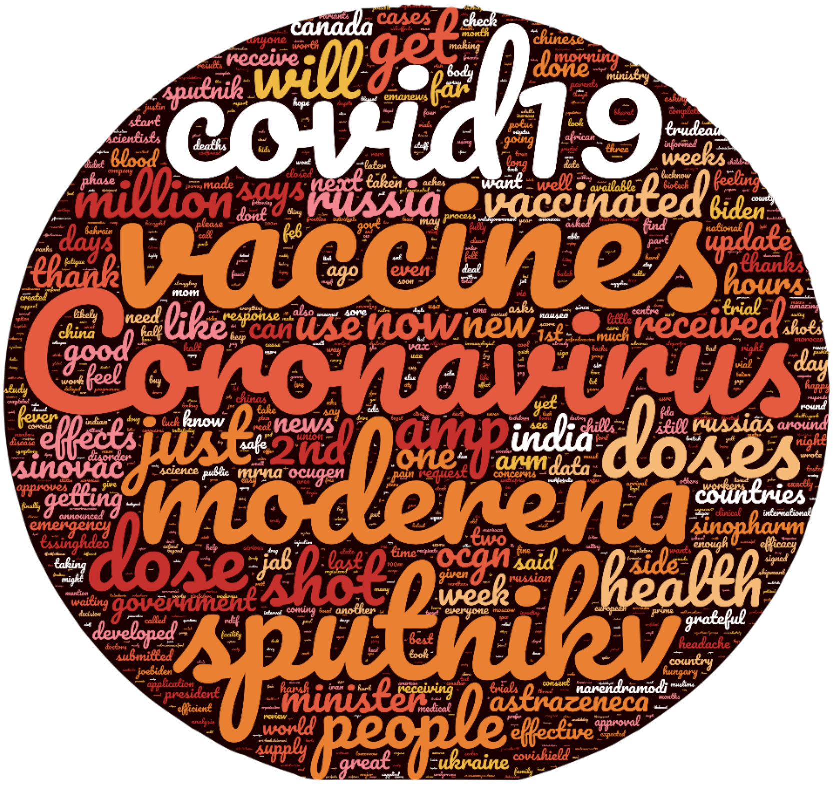


Figure 12 Word Cloud Sample

We did a statistical analysis on the Twitter vaccination dataset to see how many tweets are positive and how many tweets are negative. We plot a graph to illustrate the data.

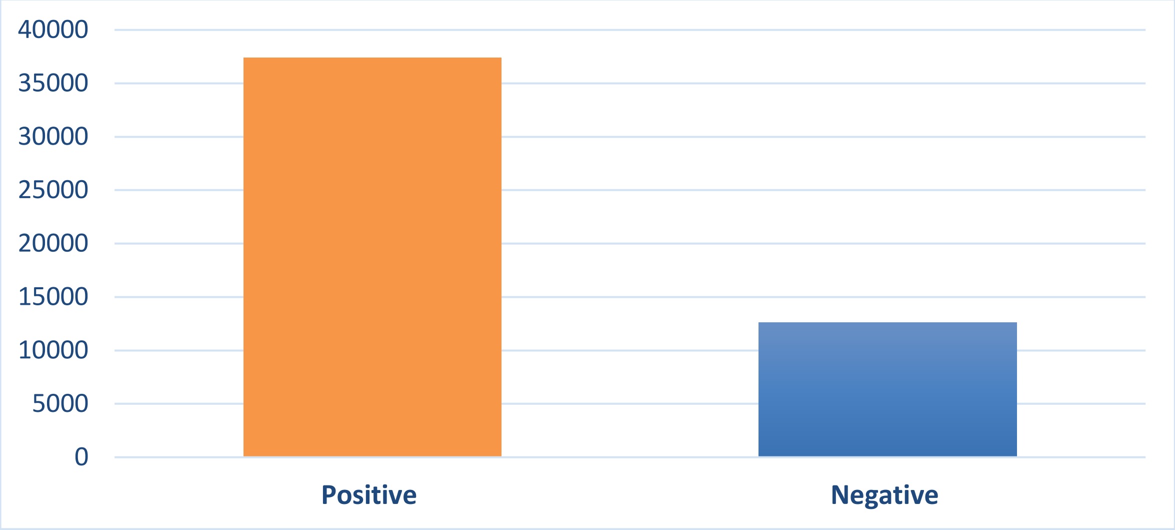


Figure 13 Statistical Analysis of Covid Vaccination Tweets

4.6 TM-BERT Model

Introduced a new modified model TM-BERT, a replica of BERT with different parameters. TMBERT is pre-trained on Wikipedia (530M) dataset which directly reduced shards time to four (4) times. We pre-trained TM-BERT model for different step sizes. We pre-trained TM-BERT model firstly on Wikipedia dataset then we add Covid-Vaccination tweets for better performance of the model on downstream tasks. We pre-trained more than one TM-BERT to achieve outperform results. We choose a distinctive combination for every trained model. As BERT model depends upon attention mechanism, the vocabulary is increased from (32K) to (52K) and a large Batch Size (1024) is used with smaller Steps (50K and 70K) along with bigger Attention Heads (24), to extract as much context as possible because TM-BERT will be fine-tuned on medical tasks. Due to the small Step size, TM-BERT is pre-trained with Learning Rate ($2e^{-5}$) and to pre-train TM-BERT deeper, the Hidden Layers are increased from (768) to (1536). To increase both, Attention Heads as well as Hidden Layers, Input Layers are decreased. Table 9 presents two (2) versions of TM-BERT with the only change in Step size. TM-BERT is pre-trained for (50K) Steps while TM-BERT_1 is pre-trained for (75K) Steps and takes (17) and (24) hours of pre-training respectively.

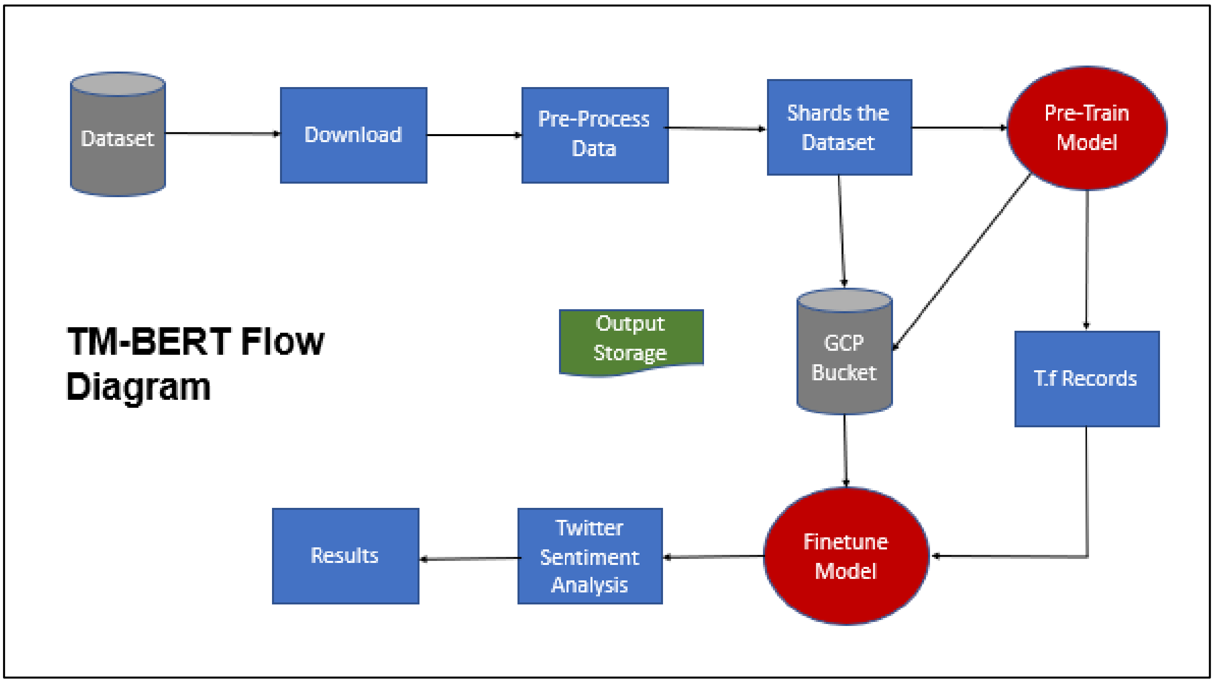


Figure 14 Flow Diagram of TM-BERT Model

4.7 Performance parameters

Multiple parameters influence the execution of the pre-trained model. For TM-BERT we use the following parameters,

4.7.1 Vocabulary Size

Vocabulary size describes the subclass of divergent tokens that can be illustrated by input. If the input contains words other than this vocabulary the model divides the word into smaller letters.

4.7.2 Sequence Size

It describes the maximum input size When one upsurges the size of the sequence it means that it requires more power to integrate resources

4.7.3 Batch Size

Batch size is the numeral of instances that can be used in a single repetition.

4.7.4 Step Size

It is measured as, how numerous steps a program will take. It takes data points concerning time.

4.7.5 Learning Rate

During the training of the model, the learning rate is hyperparameters utilized in a neural network. The weight of the learning rate can be updated during the training.

4.7.6 Hidden Layers

Hidden layers are found between the input and output layers where the operator operates the weights during execution. The most hidden layers we used, the deeper model we found.

4.7.7 Input Layers

There are hidden layers present in the transformer. Input layers are built with a neuron that delivers data to the system for first processing and then sends it for further processing in encrypted layers.

4.7.8 Attention Heads

In the transformer models each layer, there is a multiple attention head.

4.7.9 Intermediate-Size

Intermediate-size is also named as Feed-forward. 4H Feed-forward size is standardized.

4.7.10 Maximum-Position-Embeddings

Maximum position embedding is defined as the maximum sequence that the model can ever have.

4.8 Summary

In this chapter we depict the working of the BERT model in detail, BERT is a new language representation model that is pre-trained on millions of sentence pairs from Wikipedia, Bookscorpus, and many other domains and language-specific datasets. BERT is fine-tuned on the specific downstream task to attain futuristic results. At first, BERT was pre-trained on

Wikipedia(2500M) due to the unavailability of BooksCorpus dataset. A new COVID-19 Vaccination Sentiment Analysis Task (CV-SAT) is developed to finetune the BERT model to perform Sentiment Analysis on COVID-19 Vaccination Tweets. An unsupervised dataset is created using Covid Vaccination tweets. Twitter tweets about COVID19-Vaccination are gathered by using an API that contains some specific tokens, i.e., emails, URLs, emojis, emoticons, hashtag-words, and special characters like mentions. The data is then pre-processed, and the tweets are generated into meaningful sentences. Due to the BERT model's poor fine-tuning results on CV-SAT, an unsupervised Twitter dataset (70K) about COVID19-Vaccination is used to pre-train the BERT_1 model from scratch along with Wikipedia (2500M). A new Twitter Modified BERT model is proposed, named TM-BERT. TM-BERT pre-trained with different parameters on Wikipedia(530M)+ Twitter Tweets (70K) and outperforms the result from previous BERT models while pre-training on 4 times smaller dataset. And in the last performance parameters are defined, which we modified in our proposed model.

Chapter 5

Implementation and Results

CHAPTER 5: IMPLEMENTATION AND RESULTS

5.1 Implementation

We implemented our models on Google Colaboratory, using T4 of TPUs. Google Colaboraory is a free source presented jupyter notebook service that needs no set up to use. We implemented the BERT following the same parameters of the original BERT and TM-BERT by modifying the hyperparameters of the original BERT. We used the publicly available dataset of Wikipedia to pre-train the model from scratch as BookCorpus is not publicly available. We grab tweets from Twitter, related to Covid vaccination, and the pre-train model from scratch on Wikipedia+Tweets. Previously, the original BERT pre-training time took 64 hours, we reduce the time to 17 hours and lessen the computational resources of the model. We saved time as well as computational resources. For TM-BERT we deploy multiple changes in parameters like we increased the vocabulary size from 32K to 52K, and so we got more data in vocabulary. BERT depends on the attention mechanism very much, we enhance the attention head from 12 to 24, so our attention will be upgraded, and the framework is also enhanced. Increased attention head is necessary, as our data is related to the medical field, and we need more context of the sentence. We need the context of the sentence for our downstream task of sentiment analysis. We enhanced the learning rate to 2^{e-5} so that our model would learn in large step sizes. We check there is an effect of batch size and input size. As in a bigger batch size, we covered more samples, also as we enhanced attention head and attention mechanism. So, we increase the batch size from 256 to 1024 to get more and more context in one loop. We reduce the input layers from 12 to 6 as it does not affect the performance, but it consumes more resources. We enhance the hidden layers from 768 to 1536. We enhance the batch size, from which we have the bigger sample, and we reduce the learning rate. That's why there is no need to train TM-BERT up to 1M step size and we trained model up to 50K step size. This helped a lot to save time.

5.2 Pre-Training of BERT

Due to the publicly unavailability of BookCorpus dataset, BERT is pre-trained from scratch only on Wikipedia (2100M) and used SentencePiece instead of WordPiece. BERT is pre-trained for (64) hours with the hyper-parameters presented in Table 4

Table 4 Pre-training Hyper-parameters

Model	Step	Batch Size	Input Layers	Hidden layers H	Attention Heads	Maximum Position Embedding	Feed Forward	Learning Rate	Vocabulary
BERT	1M	256	12	768	12	512	3072 (4H)	1e^{-5}	32000

5.3 Fine-Tuning of BERT

BERT is fine-tuned on COVID-19 Vaccination Sentiment Analysis Task (CV-SAT). It achieved (0.70) accuracy on CV-SAT with fine-tuning parameters which are Batch Size (32), Maximum Sequence Length (128) and Learning Rate (2e-^{5}).

Reduced the time from 64H to 17H. We took step size to 75K, increasing the step size also increased the completion time but did not improve the performance very much. The performance increases a little bit, but this small increase in performance takes too much time. As it takes, almost 25 hours to complete the task. Results are very much improved by changing parameters and pre-training time is reduced by 4 times. This consumes fewer resources than the original BERT model fine-tuning process.

5.4 Dataset for Pre-Training

BERT has achieved (70%) accuracy by fine-tuning on CV-SAT but as affirmed by Lee, J et al.[23] and Beltagy, I et al.[24] that pre-training BERT on domain-specific datasets increases its performance on the downstream task, a domain-specific unsupervised dataset about COVID-19 Vaccination is then developed. Twitter tweets (70K) about COVID19-Vaccination are gathered by using an API that contains some specific tokens, i.e., emails, URLs, contact numbers, mathematical numbers, pricing, different time formats, emojis, emoticons, hashtag-words, and special characters like mentions. The data is then pre-processed, and the tweets are generated into meaningful sentences and formal forms.

The dataset consisting (70K) tweets is merged twice; firstly, with Wikipedia (2100M) (2.5 B) words to pre-train BERT_1 model, as shown in Table 5, and secondly with (530M) Wikipedia dataset to pre-train the newly proposed TM-BERT, as shown in Table 6

Table 5 BERT_1 Pre-Training Dataset

Corpus	**Number of lines**	**Domain**
English Wikipedia	2100M	General
Tweets	70 K	Covid Vaccination

Table 6 TM-BERT Pre-Training Dataset

Corpus	**Number of lines**	**Domain**
English Wikipedia	530M	General
Tweets	70 K	Covid Vaccination

5.5 Pre-Training of BERT_1

BERT_1 is pre-trained for (64) hours on the dataset mentioned in Table 5 and with the hyper-parameters presented in Table 4 with only a single alteration of the Learning Rate being (2e^{-5}), which occurred due to the increase in the size of the pre-training dataset.

5.6 Fine-Tuning of BERT_1

BERT_1 is fine-tuned on CV-SAT and achieved (0.76) accuracy, a (6%) improvement with the similar fine-tuning parameters as those of BERT.

Table 7 Hyper-parameters for BERT_1 pre-training on Wikipedia + Tweets

Model	**Step**	**Batch Size**	**Input Layers**	**Hidden Layers H**	**Attention Heads**	**Maximum Position Embedding**	**Feed Forward**	**Learning Rate**	**Vocabulary**
BERT_1	1M	256	12	768	12	512	3072(2H)	2e-5	32000

5.7 Pre-Training of TM-BERT

TM-BERT (Twitter-Modified BERT) is pre-trained on the dataset mentioned in Table 6 for (17) hours. TM-BERT model is pre-trained by modifying hyper-parameters during pre-training to increase the overall performance on CV-SAT. To modify BERT for improved performance on CV-SAT, this study follows the findings of Shah et al. [5] which proclaimed that models like BERT do not require to be pre-trained on large general datasets like Wikipedia (2100M). Based on this, TMBERT is pre-trained on Wikipedia (530M) dataset which directly reduced shards time to four (4) times.

Table 8 Hyperparameters of TM-BERT

Hyperparameter	Values
Step Size	50,000
Batch size	1024
Input layers	6
Hidden layers	1536
Attention heads	24
Maximum position encoding	512
Vocabulary	52,000
Learning rate	2e-5
Maximum Position Embeddings	512
Masking	15%
Feed Forward	3072(2H)
Time	17 Hours

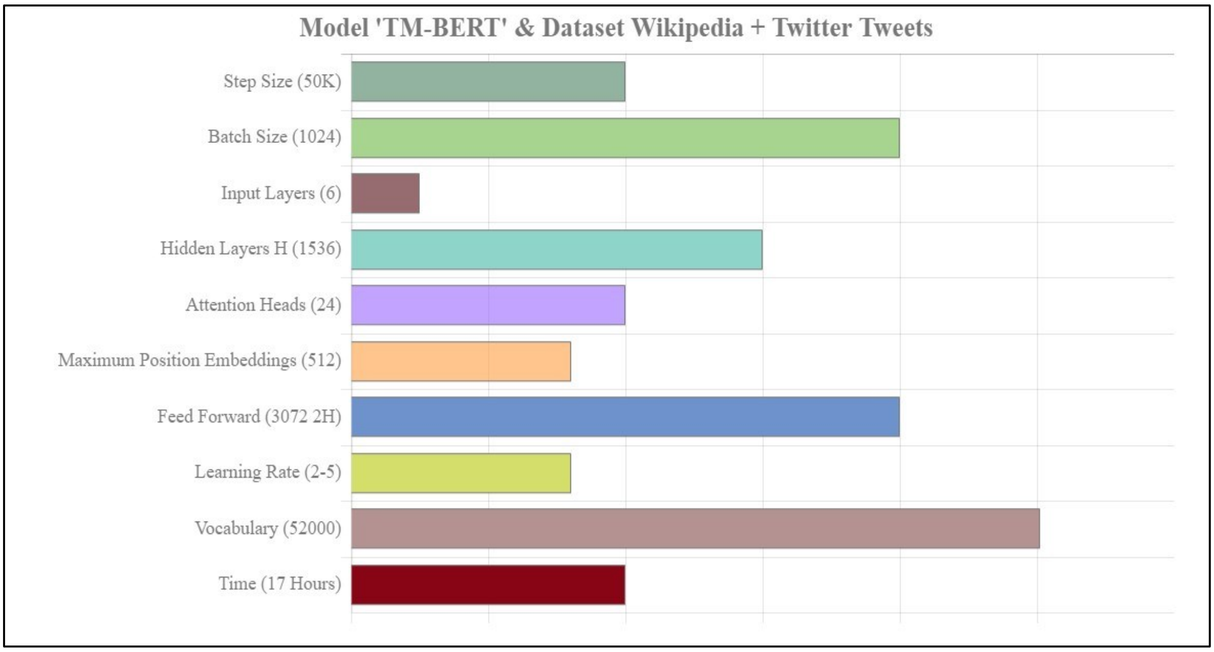


Figure 15 TM-BERT Hyperparameters

5.8 Fine-Tuning of TM-BERT

Fine-Tuned on Google Colaboratory using TPU. A batch size of 16 was selected. The learning rate of 2e-5 was Selected. TM-BERT outperformed on Twitter datasets. Fine-tuned results of TM-BERT give much better and different results of Twitter sentiment analysis from the previously performed analysis. We fine-tuned TM-BERT model on the downstream task of the Covid Vaccination Sentiment Analysis Task (CV-SAT). TM-BERT outperforms on the downstream task of sentiment analysis. We changed parameters as the batch size of 16, the maximum sequence length of 128, and a learning rate of 2e-5. TM-BERT's pre-trained model is fine-tuned on the CV-SAT task. TM-BERT and TM-BERT_1 achieved (0.89) and (0.90) accuracy with fine-tuning hyperparameters of Batch Size of (16), maximum Sequence Length of (128), and a Learning Rate of (2e-5).

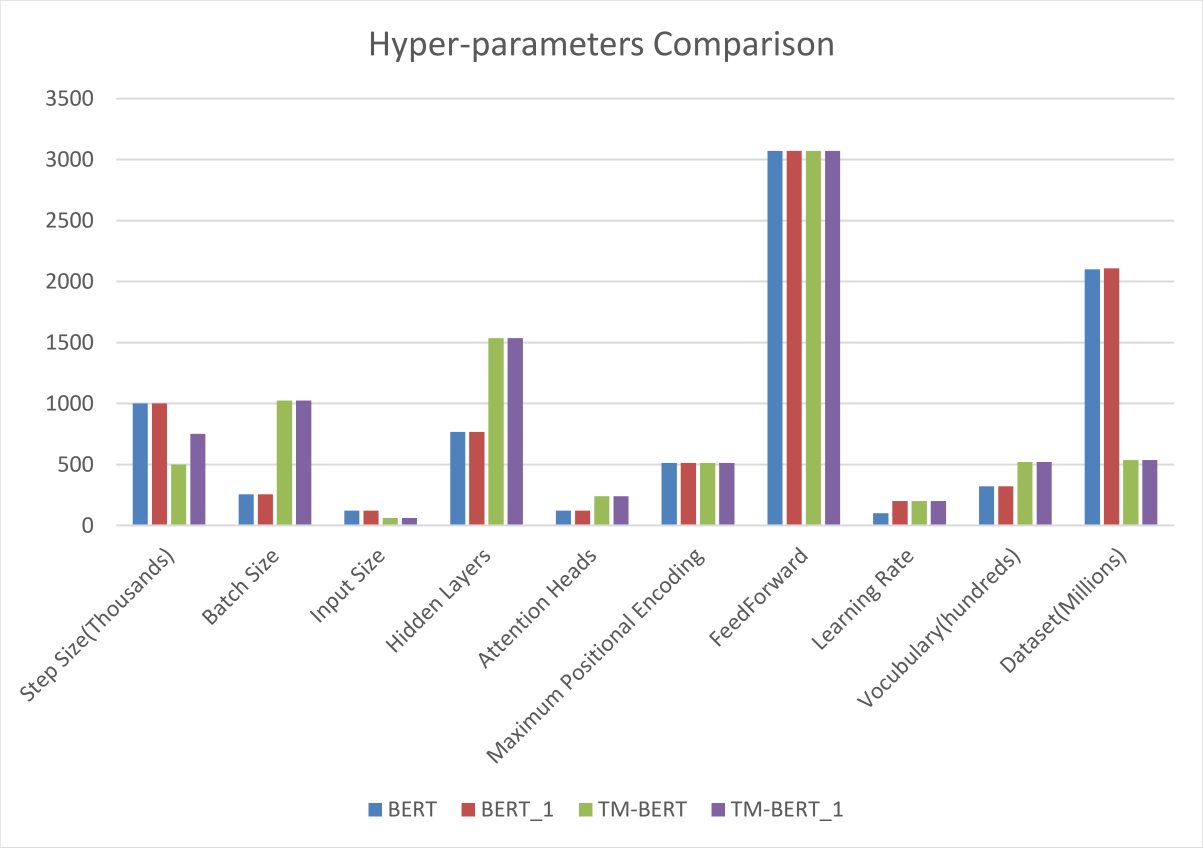


Figure 16 Hyper-parameters Comparison

Table 9 All Pre-Trained Models

Model	**Step**	**Batch size**	**Input Layers**	**Hidden layers H**	**Attention Heads**	**Maximum Position Embedding**	**Feed Forward**	**Learning Rate**	**Vocabulary**	**Dataset**
BERT	1M	256	12	768	12	512	3072 (4H)	1e^{-5}	32000	Wiki
BERT_1	1M	256	12	768	12	512	3072 (4H)	2e^{-5}	32000	Wiki + Tweets
TMBERT	50K	1024	6	1536	24	512	3072 (2H)	2e^{-5}	52000	Wiki+ Tweets
TMBERT_1	75K	1024	6	1536	24	512	3072 (2H)	2e^{-5}	52000	Wiki+ Tweets

5.9 Results

We pre-trained and fine-tuned both models BERT and TM-BERT on Wikipedia and Wikipedia + Covid Vaccination Tweets, but here in the result section we only mention fine-tuned results of Wikipedia + Covid Vaccination Tweets. Both models outperform Twitter sentiment analysis tasks. We fine-tuned all the pre-trained models on the sentiments analysis task, and we achieve good results from previous models only just by doing modifications in the parameters of the original BERT. Also, the increased sized vocabulary helps in the understanding of the language more.

This study pre-trained BERT on Wikipedia (2100M) and BERT_1 on the dataset mentioned in Table 2 and then fine-tuned on newly developed CV-SAT. The study also proposed a new TM-BERT model pre-trained on the dataset mentioned in Table 3 and fine-tuned it on CV-SAT. Figure 2 presents the results of BERT, BERT_1, TM-BERT, and TM-BERT_1 while fine-tuning on CV-SAT. The performance achieved by the models on CV-SAT is BERT (0.70), BERT_1 (0.76), TM-BERT (0.89) and TM-BERT_1 (0.90). TM-BERT outperformed BERT with (19%) and BERT_1 with (13%) accuracy while TM-BERT_1 outperformed BERT with (20%), BERT_1 with (14%) and TM-BERT with (1%) accuracy.

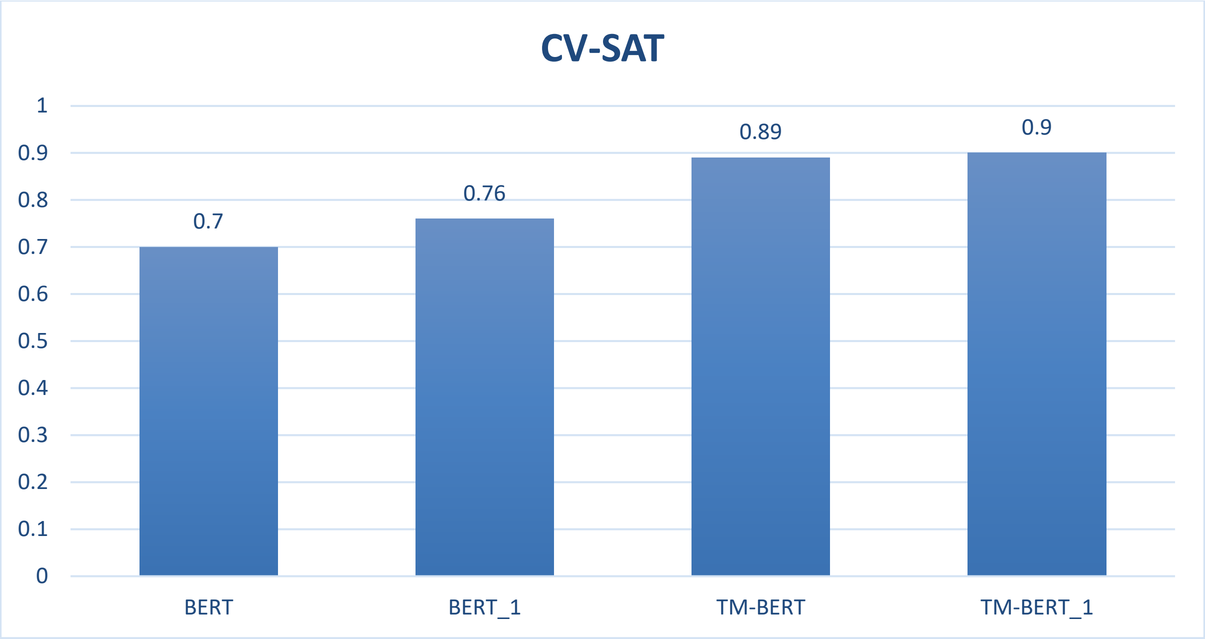


Figure 17 Final Results

5.10 Summary

We implemented 4 different models on Google Colaboratory, using T4 of TPUs. First of all, due to the public unavailability of BookCorpus dataset, BERT is pre-trained from scratch only on Wikipedia (2100M) and used SentencePiece instead of WordPiece. BERT is pre-trained for (64) hours with the same hyper-parameters used by the original BERT. BERT_1 is pre-trained for (64) hours on Wikipedia (2100M) and Twitter Tweets (70K) and with the hyper-parameters presented for the BERT with only a single alteration of the Learning Rate being ($2e^{-5}$), which occurred due to the increase in the size of pre-training dataset. TM-BERT (Twitter-Modified BERT) is pre-trained on Wikipedia (530M) and Twitter Tweets (70K) for (17) hours with a step size of 50K. TM-BERT model is pre-trained by modifying hyper-parameters during pre-training to increase the overall performance. Similarly, TM-BERT_1 is pre-trained on Wikipedia (530M) and Twitter Tweets (70K) for (25) hours with a step size of 75K. All the BERT, BERT_1, TM-BERT and TM-BERT_1 model is fine-tuned on the downstream task of the Covid Vaccination Sentiment Analysis Task (CV-SAT).

The performance achieved by the models on CV-SAT is BERT (0.70), BERT_1 (0.76), TM-BERT (0.89) and TM-BERT_1 (0.90).

Chapter 6

Discussion, Conclusion and Future Work

CHAPTER 6: DISCUSSION, CONCLUSION AND FUTURE WORK

During the peak days of Covid-19, people are scared of this infectious disease. People are restricted to their places of shelter. In such dire times of global pandemic, Twitter proved to be a vital source for getting reliable information about the COVID-19 situation worldwide. We check people's tweets regarding this COVID-19 outbreak and we decided to get to know people's behavior or sentiment for this outbreak. We decided to perform sentiment analysis on Twitter tweets, as Twitter's text format is perfect to make sentiment analysis datasets. When we merged tweets collected from Twitter in one corpus, it becomes very useful information which is also very helpful to understand the social sentiment of people regarding a particular element. We choose to perform sentiment analysis on Covid Vaccination tweets. As COVID-19 vaccination was made mandatory for every individual to counter the virus, people from developed countries like the USA and the UK, as well as from developing countries like Pakistan, gave mixed reactions towards the vaccination.

In this work, a new COVID-19 Vaccination Sentiment Analysis Task (CV-SAT) is developed to analyze the sentiments of people about COVID-19 Vaccination. An unsupervised COVID-19 Vaccination dataset is also developed for the pre-training of language models built upon Transformer architecture. This Twitter pre-training dataset contains clean and preprocessed (70K) tweets about Covid-19 Vaccination.

BERT is a pre-trained unsupervised language model that was proposed by Google AI research team. It took first place on a wide variety of Natural Language Processing tasks. In this paper, based on several settings of BERT model and different hyperparameters of it, we derive our own BERT model (TM-BERT), which is optimized for NLP tasks. The performance of TM-BERT models are evaluated on Sentiment Analysis task. The results of the evaluation method show the superiority of our models. We pre-trained the BERT model from scratch with original parameters and replicate the original BERT model by changing the hyperparameters. This proposed replica model is named TM-BERT. We pre-trained the BERT model only on the Wikipedia dataset due to the publicly unavailability of BookCorpus dataset. The hyperparameters of BERT remain the same but with a single change of using WordPiece embedding instead of using SentencePiece. BERT is pre-trained on Wikipedia (2100M) whereas BERT_1 is pre-trained on Wikipedia (530M) + Tweets(70K) and fine-tuned on CV-SAT task and achieved (0.70) and (0.76) accuracy respectively while pre-training for (64) hours.

This study proposed a modified BERT model called TM-BERT to pre-train BERT on smaller datasets with a lesser pre-training time and enhance the overall performance of BERT with (19%). The study pre-trained two versions of TM-BERT with changed Step sizes of (50K) and (75K). TM-BERT_1 which pre-trained for (75K) steps achieved (1%) accuracy over TM-BERT but took (50%) longer pre-training time, (24) hours instead of (17) hours for TM-BERT (50K) steps. Improvement of (1%) for (50%) of more training is negligible so therefore this study proposed a TM-BERT model with just (17) hours of pre-training and (19%) improved accuracy over BERT. BERT model was pre-trained for the first time on the Covid Vaccination task and also performed sentiment analysis on the CV-SAT dataset. That is why we have not compared it with any other model that has been pre-trained for the covid vaccination task.

In this paper, the shards time needed to pre-train the BERT model is ignored, if we considered this time also, this makes the pre-training of TM-BERT directly four (4) times less because TM-BERT is pre-trained on four (4) times smaller datasets. The future direction of this study is to implement the TM-BERT model in real-time systems to provide real-time results about people's behavior against every new wave of COVID-19 and its booster doses. And we implement this TM_BERT model for any new biomedical domain.

CONTRIBUTIONS

- Developed a new COVID-19 Vaccination Sentiment Analysis Task (CV-SAT)
- Developed a new unsupervised domain specific COVID-19 Vaccination Twitter Tweets (70K) dataset for pre-training.
- Proposed a TM-BERT model that significantly reduces the pre-training from (64) hours to just (17) hours with a (19%) improvement.
- Pre-trained BERT on Wikipedia (2100M) from scratch and fine-tuned on CV-SAT.
- Pre-trained BERT_1 on Wikipedia (2100M) + COVID-19 Vaccination Twitter Tweets (70K) dataset and fine-tuned on CV-SAT..
- Using limited resources results are improved from BERT model and previously pre-trained models which were pre-trained on Twitter tweets.

REFERENCES

[1] A. Alamoodi *et al.*, "Sentiment analysis and its applications in fighting COVID-19 and infectious diseases: A systematic review," vol. 167, p. 114155, 2021.

[2] K. H. Manguri, R. N. Ramadhan, and P. R. M. J. K. J. o. A. R. Amin, "Twitter sentiment analysis on worldwide COVID-19 outbreaks," pp. 54-65, 2020.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. J. a. p. a. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2018.

[4] N. Azzouza, K. Akli-Astouati, and R. Ibrahim, "Twitterbert: Framework for twitter sentiment analysis based on pre-trained language model representations," in *International Conference of Reliable Information and Communication Technology*, 2019, pp. 428-437: Springer.

[5] M. Shah Jahan, H. U. Khan, S. Akbar, M. Umar Farooq, S. Gul, and A. J. S. P. Amjad, "Bidirectional Language Modeling: A Systematic Literature Review," vol. 2021, 2021.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805,* 2018.

[7] X. Yu, W. Hu, S. Lu, X. Sun, and Z. Yuan, "Biobert based named entity recognition in electronic medical record," in *2019 10th international conference on information technology in medicine and education (ITME)*, 2019, pp. 49-52: IEEE.

[8] A. M. Dai and Q. V. J. A. i. n. i. p. s. Le, "Semi-supervised sequence learning," vol. 28, 2015.

[9] J. Howard and S. J. a. p. a. Ruder, "Universal language model fine-tuning for text classification," 2018.

[10] B. McCann, J. Bradbury, C. Xiong, and R. Socher, "Learned in translation: Contextualized word vectors," in *Advances in Neural Information Processing Systems*, 2017, pp. 6294-6305.

[11] B. McCann, J. Bradbury, C. Xiong, and R. J. A. i. n. i. p. s. Socher, "Learned in translation: Contextualized word vectors," vol. 30, 2017.

[12] G. Lample and A. J. a. p. a. Conneau, "Cross-lingual language model pretraining," 2019.

[13] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding with unsupervised learning," 2018.

[14] L. Dong *et al.*, "Unified language model pre-training for natural language understanding and generation," vol. 32, 2019.

[15] Y. Sun *et al.*, "Ernie: Enhanced representation through knowledge integration," 2019.

[16] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. J. a. p. a. Liu, "Mass: Masked sequence to sequence pre-training for language generation," 2019.

[17] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. J. A. i. n. i. p. s. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," vol. 32, 2019.

[18] J. Sarzynska-Wawer *et al.*, "Detecting formal thought disorder by deep contextualized word representations," vol. 304, p. 114135, 2021.

[19] A. Vaswani *et al.*, "Attention is all you need," vol. 30, 2017.

[20] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. J. a. p. a. Soricut, "Albert: A lite bert for self-supervised learning of language representations," 2019.

[21] Y. Liu *et al.*, "Roberta: A robustly optimized bert pretraining approach," 2019.

[22] H. Lee, J. Yoon, B. Hwang, S. Joe, S. Min, and Y. Gwon, "Korealbert: Pretraining a lite bert model for korean language understanding," in *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021, pp. 5551-5557: IEEE.

[23] J. Lee *et al.*, "BioBERT: a pre-trained biomedical language representation model for biomedical text mining," vol. 36, no. 4, pp. 1234-1240, 2020.

[24] I. Beltagy, K. Lo, and A. J. a. p. a. Cohan, "SciBERT: A pretrained language model for scientific text," 2019.

[25] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. J. T. o. t. A. f. C. L. Levy, "Spanbert: Improving pre-training by representing and predicting spans," vol. 8, pp. 64-77, 2020.

[26] Y. Sun *et al.*, "Ernie 2.0: A continual pre-training framework for language understanding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, vol. 34, no. 05, pp. 8968-8975.

[27] J. Wei *et al.*, "NEZHA: Neural contextualized representation for chinese language understanding," *arXiv preprint arXiv:1909.00204,* 2019.

[28] J. Wei *et al.*, "Nezha: Neural contextualized representation for chinese language understanding," 2019.

[29] W. Wang *et al.*, "Structbert: Incorporating language structures into pre-training for deep language understanding," 2019.

[30] X. Jiao *et al.*, "Tinybert: Distilling bert for natural language understanding," 2019.

[31] X. Liu, P. He, W. Chen, and J. J. a. p. a. Gao, "Multi-task deep neural networks for natural language understanding," 2019.

[32] K. Clark, M.-T. Luong, Q. V. Le, and C. D. J. a. p. a. Manning, "Electra: Pre-training text encoders as discriminators rather than generators," 2020.

[33] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. J. A. i. N. I. P. S. Zhou, "Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers," vol. 33, pp. 5776-5788, 2020.

[34] X. Liu, P. He, W. Chen, and J. J. a. p. a. Gao, "Improving multi-task deep neural networks via knowledge distillation for natural language understanding," 2019.

[35] C. Raffel *et al.*, "Exploring the limits of transfer learning with a unified text-to-text transformer," 2019.

[36] C. Zhu, Y. Cheng, Z. Gan, S. Sun, T. Goldstein, and J. J. a. p. a. Liu, "Freelb: Enhanced adversarial training for natural language understanding," 2019.

[37] D. Chen *et al.*, "Adabert: Task-adaptive bert compression with differentiable neural architecture search," 2020.

[38] M. Müller, M. Salathé, and P. E. J. a. p. a. Kummervold, "Covid-twitter-bert: A natural language processing model to analyse covid-19 content on twitter," 2020.

[39] M. Pota, M. Ventura, R. Catelli, and M. J. S. Esposito, "An effective BERT-based pipeline for Twitter sentiment analysis: a case study in Italian," vol. 21, no. 1, p. 133, 2020.

[40] A. Karimi, L. Rossi, and A. Prati, "Adversarial training for aspect-based sentiment analysis with BERT," in *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021, pp. 8797-8803: IEEE.

[41] H. Xu, B. Liu, L. Shu, and P. S. J. a. p. a. Yu, "BERT post-training for review reading comprehension and aspect-based sentiment analysis," 2019.

[42] W. de Vries, A. van Cranenburgh, A. Bisazza, T. Caselli, G. van Noord, and M. J. a. p. a. Nissim, "Bertje: A dutch bert model," 2019.

[43] X. Dai, S. Karimi, B. Hachey, and C. J. a. p. a. Paris, "Cost-effective selection of pretraining data: A case study of pretraining BERT on social media," 2020.

[44] M. Singh, A. K. Jakhar, S. J. S. N. A. Pandey, and Mining, "Sentiment analysis on the impact of coronavirus in social life using the BERT model," vol. 11, no. 1, pp. 1-11, 2021.

[45] S. Palani, P. Rajagopal, and S. J. a. p. a. Pancholi, "T-BERT--Model for Sentiment Analysis of Micro-blogs Integrating Topic Model and BERT," 2021.

[46] A. Z. Klein, A. Magge, K. O'Connor, J. I. F. Amaro, D. Weissenbacher, and G. G. J. J. o. m. I. r. Hernandez, "Toward using Twitter for tracking COVID-19: a natural language processing pipeline and exploratory data set," vol. 23, no. 1, p. e25314, 2021.

[47] D. Q. Nguyen, T. Vu, and A. T. J. a. p. a. Nguyen, "BERTweet: A pre-trained language model for English Tweets," 2020.

[48] M. Pota, M. Ventura, H. Fujita, and M. J. E. S. w. A. Esposito, "Multilingual evaluation of pre-processing for BERT-based sentiment analysis of tweets," vol. 181, p. 115119, 2021.

[49] J. A. Gonzalez, L.-F. Hurtado, and F. J. N. Pla, "TWilBert: Pre-trained deep bidirectional transformers for Spanish Twitter," vol. 426, pp. 58-69, 2021.

[50] Y. Liu *et al.*, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692,* 2019.

[51] E. Loper and S. J. a. p. c. Bird, "Nltk: The natural language toolkit," 2002.

[52] F. Heimerl, S. Lohmann, S. Lange, and T. Ertl, "Word cloud explorer: Text analytics based on word clouds," in *2014 47th Hawaii international conference on system sciences*, 2014, pp. 1833-1842: IEEE.