Sentimental Analysis On Covid-19 Tweets using Bidirectional Encoder Representations Transformers

Deekshitha J P*, Rakshitha Shankar†, Shantala C P‡, Girish L§
CIT Open Source Club, Department of Computer Science & Engineering
Channabasaveshwara Institute of Technology, Gubbi, Karnataka, India
Email: *deekshithajp@gmail.com, †rakshitha.shivashankar@gmail.com, ‡shan1675@gmail.com, §girishlingappa7@gmail.com

***Abstract*—The Coronavirus epidemic has wreaked havoc on countries all over the world. People from all over the Globe have flocked to social network to voice their thoughts and feelings about the situation that has reached epidemic proportions. The need of this paper is to exemplify how social media users feel about COVID-19 in an extremely brief span of time, Twitter saw an unusual surge in tweets on the new Coronavirus. This study presents a global Sentiment Analysis of tweets related to COVID-19, as well as how people's sentiment in multiple nations has varied. The research study focuses on a period of time from march 2020 to april 2020. In the Sentiment Analysis, we fed dataset to different algorithms and estimate the best performance among them.As in secondly we also found the reliability on BERT model. Comparatively, BERT gave foremost accuracy amidst all. Besides, the accuracy of mentioned algorithms are well represented.**

***Index Terms*—Coronavirus, Twitter, Natural Language Processing, Sentiment Analysis, Machine Learning, BERT.**

I. Introduction

The novel Coronavirus pandemic is the massive problem facing all over the world and it has had effected many lives of human beings [1]. COVID-19 is a newly identified coronavirus that causes an infectious sickness. The disease caused chaos on people's physical well being, financial environment, conditions of employment, and manufacturing industries [2]. Due to pandemic the lifestyle of the people revolutionized. The role of social networks is getting increasingly prominent than it has ever been. Twitter and other social media sites are excellent resources for capturing human feelings and thoughts.

Over this tough time, many have turned to express their concerns, viewpoints, and perspectives about the world situation on social media. This research focuses on the evaluation of tweets[2]. Probing tweets now and succeeding Coronavirus could be notable because the circumstances and folk's emotions are altering all over the time[3]. To determine the tweets' sentimental content, a tactic known as Sentiment Analysis was employed to identify them as Positive, Negative, or Neutral. Sentiment Analysis (also known as opinion mining or emotion AI) is a branch of natural language processing that attempts to recognise and collect personal views from a particular text, such as blog posts, reviews, or social networking sites[4].

NLP is a branch of computer science that makes extensive use of machine learning and computational rhetoric. The goal of this field is to make human-computer interaction simple but effective[5]. The concept of word embedding is the brains behind NLP. Word embeddings are nothing more than words in the form of vectors, which are learned by analysing large amounts of text, when used as the underlying input representation will boost the performance in NLP tasks.This field is primarily concerned with making human-computer interaction uncomplicated but impactful.

Several NLP tasks can help the machine understand what it's taking by simplifying human speech and audio. Speech recognition (converts audio input to textual data), Part of speech tagging (determines the part of speech of a given word or piece of content depending on its use and situation), Named entity recognition (recognises specific words or phrases as usable entities) and Sentiment analysis (for extracting subjective aspects from text, such as attitudes, emotions, sarcasm, bewilderment, and suspicion) are just a few of the tasks available.

Some popular NLP tools are - **CoreNLP** from Stanford group. The most well-known NLP library for Python is **NLTK**. **TextBlob** is an NLTK interface that is both user-friendly and intuitive. **Gensim** is a document similarity analysis package. **SpaCy** is a high-performance natural language processing package. **PyTorch**, is a fully configurable and extensible[6].

Natural language processing technology being one of the widely implemented fields of ML. It has the ability to understand, analyze, manipulate, and potentially generate human language. In this project, we contemplated the ample data set of tweets by internet subscribers all over the world. The choosing of Machine Learning models like SVC, Logistic Regression(LR), Naive Bayes, and Random Forest algorithms were compared and used to determine the accuracy performance with BERT. BERT (Bidirectional Encoder Representations from Transformers) is the most recent deep-learning model for textual data processing.

Our challenge is to construct a classification model to forecast the sentiment of COVID-19 tweets. The tweets are extracted.We are given details like Location, Tweet At, Original Tweet, and Sentiment.

Initially we determine the accuracy of the above models using vectorizers like Term Frequency-Inverse document Frequency(tf-idf) and Count Vectorizer. We choose best fit and also we calculate accuracy of BERT model.

Our intention is to identify and categorize view point conveyed in tweets, particularly so as to resolve even if the contributor has a positive, negative, or neutral attitude toward a primary issue.
The benefaction of this whole task is to inspect the COVID-19 tweets dataset and we compute the highest possible accuracy to find the polarity of social media tweets. In the subsequent sections, we will look at relevant works, refer to articles, deal with system models, conduct an experiment, and eventually arrive to the end of the research.

II. Related Work

The largest bulk of COVID-19 tweets are analyzed by using ML algorithms to determine sentiments. In this chapter we elaborate the concepts of Machine Learning, Sentiment Analysis and Natural Language Processing.

1) MACHINE LEARNING

ML is a subset of data science that automates the development of analytical models. It's an area of artificial intelligence and computer science premised on the notion that systems can study from data, identify patterns, and emulate the way humans learn, gradually improving their accuracy. Machine learning is now so prevalent that you may use it more than once a day without even realising it. There are a few popular ways to machine learning. Supervised Learning, Unsupervised Learning, Semi-supervised Learning, and Reinforcement Learning are the four types of learning[7].
Supervised Learning is a technique for teaching a machine using labelled data. The algorithm is given a list of inputs and their corresponding correct outcomes, and it develops by contrasting its actual performance to the proper outputs in order to detect flaws. The type of problems solved in supervised learning are regression and classification. Regression is about predicting a continuous quantity. Predicting a label or class is what classification is concerned[8].
Unsupervised Learning is used for unlabelled data. There is no guidance here,in order to make predictions about the results, the computer must first figure out the data set and hunt for hidden patterns. The idea is to look over the data and see if there is any structure within. Popular tactics include clustering and association. Clustering is the process of grouping similar entities together, whereas association involves discovering patterns in data and finding co-occurrences and so on[20].
Semi-supervised learning involves using a combination of labelled and unlabeled data for training, with a small amount of labelled data and a big number of unlabeled data[8]. Many neural network models and training methods can be combined in this way. To reduce error and enhance model performance, the goal is to develop a predictor that approximates future test data better than the predictor learned from labelled training data. Methods like classification, regression, and prediction can be used to implement this type of learning[18][19].
Reinforcement Learning The algorithm determines which acts provide the highest rewards through a trial and error process. This algorithm basically learns from experience. In this learning, the key difference is that input depends on actions we take. The whole reinforcement itself is a training and testing phase since no predefined data is given, machine itself has to learn everything on its own and it starts by exploring and connecting data. As a result, the objective of reinforcement learning is to discover the best policy.
Machine learning for natural language processing and text analysis entails a set of statistical algorithms for detecting portions of speech, entities, sentiment, and other properties of text.

2) NATURAL LANGUAGE PROCESSING

As the title clears our perception that it has a sort of processing to do with language. NLP is an intersection of Artificial intelligence, Computer Science and Linguistics. NLP examines the grammatical structure of sentences as well as the particular meanings of words, then use algorithms to extract meaning and provide outputs. The end goal of this technology is for computers to understand the content, nuances(subtle differences) and the sentiment of the document. NLP is a tricky problem since human language is imprecise.
Syntactic and semantic analysis are two techniques used in Natural Language Processing (NLP), it is a technique used to assist machines in interpreting text. Tokenization (splitting up an up into discrete parts called tokens), part of speech tagging (labelling tokens as verb, adverb, adjective, noun, etc. to help deduce the meaning of the text), Lemmatization and Stemming(lowering inflected words to their ground form to facilitate analysis), and Stop Word removal(removing high-frequency terms from a sentence that have little or no semantic significance) are all examples of syntactic analysis [9].
Some Natural Language Processing algorithms that we are considering in this paper -**SVC** is a non-parametric clustering technique that makes no implications about the size or form of the data clusters[10], **Naïve Bayes Classifier**is a suitable and efficient classification approach that assists in the building of fast machine learning models that can make accurate predictions quickly. The Bayes theorem is often used. To predict the likelihood of a target attribute, a supervised learning classification approach known as **Logistic Regression** is used. Data is entered as 1 (represents success/yes) or 0 (represents failure/no) for the dependent variable. For both classification and regression, **Random Forest Classifier** is used. It builds decision trees from randomly chosen sample data, gets recommendations from each tree, and then votes on the finest option [11].
Further addition of vectorizer to the above algorithms can speed up the python code without any loops and obtain the optimization. In our paper, we'll use TF-IDF and Count Vectorizer. **Term frequency-inverse document frequency** (tf-idf) is a measure that considers the importance of a term based on how often it appears in a document and corpus. The TF-IDF is calculated as follows:
tf('word') = number of times 'word' appears in document / total number of words in document.
idf('word') = log(total number of documents / number of

documents containing the word 'word')[12].
Multipying the tf and idf numbers results in the TF-IDF score of a word in a corpus.
Count Vectorizer(CV) transforms a phrase into a vector by counting the number of occurrences of each word in the text. A column in the matrix denotes each unique word, while a row in the matrix denotes each text sample from the corpus. The value of each cell represents the number of words in the text sample [13].

3) SENTIMENT ANALYSIS

The problem with sentiment analysis is categorized as supervised machine learning. Sentiment analysis is a technique for detecting emotions in text and classifying them as positive, negative, and neutral. Companies may acquire insight into how customers feel about brands and goods by studying social media posts, product reviews, and online surveys. You may figure out which areas of your customer service earn positive or negative feedback by evaluating open-ended replies to NPS questionaires. There are different sorts of sentiment analysis, which include the following:
Fine-Grained Sentiment Analysis (A sentence is broken down into phrases or sentences, each of which is examined in relation to the others.)
Aspect-Based Sentiment Analysis (Customer feedback is analysed by correlating specific attitudes with various product or service aspects.) [14]
Emotion Detection (compiles data on a person's verbal and nonverbal communication to assess their mood or attitude) [1].
Table 1 summarises all of the papers we used in our investigation. To demonstrate how sentiment analysis works, there are three basic processes-
Data Collection: This requires the usage of certain keywords or hashtags to match the content that users are looking for based on their interests. This info comes in a wide variety of formats (e.g. tweets, posts, news, texts).
Prepossessing: This step entails fine-tuning the information gathered in order to prepare it for the following step. This phase consists of three major steps: The cleaning step includes tasks such as repetitive letter removal, text correction, normalisation, stop word removal, and language detection. After then, the Tokenization technique converts text to tokens till vectors are created. Finally, the mining features and grammatical structures are retrieved.
Data analysis: All data should be analysed and identified at this stage based on the primary goal of the study, such as polarity identification, sentiment analysis, or frequency analysis. The main goals of the study was determined, such as polarity detection, sentiment analysis, or frequency analysis [3]. In the next segment, we'll go through the referred papers which will assist us with our research.

III. EXISTING WORK

Muvazima Mansoor [2] narrated that the essence was built on a Covid-19 dataset of English tweets (567,064 tweets were processed and analysed). The latent dirichlet allocation method was used to extract topics, and a lexicon-based strategy was used to conduct sentiment analysis. The researchers used the Spark platform in combination with Python to improve and optimise the analysis and processing of huge amounts of social data. This work was lemmatized three times to get the best results and tackle the issues of translating words to their correct roots.
Manal Abdulaziz [3] research looks at global sentiment analysis of Coronavirus tweets, as well as how people's sentiments have changed over time in different nations. Over time, there was a shift in public opinion. The accuracy of various Machine Learning models for sentiment categorization, such as Long Short Term Memory (LSTM) and Artificial Neural Networks (ANN), was also identified. An exploratory data analysis is also done on a dataset containing daily data on the number of cases reported. Positive and negative views, fear, and trust emotions expressed in tweets were all examined in the resulting datasets. On the datasets for Sentiment Classification, two deep learning methods (LSTM and ANN) were used.
Nalini Chintalapudi [17] explained that the performance of the Bidirectional Encoder Representations from Transformers (BERT) model was compared to three different models: logistic regression (LR), support vector machines (SVM), and long-short term memory (LSTM). Each sentiment's precision was evaluated separately. The accuracy of the BERT model was 89 percent, compared to 75, 74.7, and 65 percent for the other three models. According to these figures, keywords and related terms were used often in Indian tweets during COVID-19. Furthermore, this study clarifies public perceptions of epidemics and directs public health professionals towards a better society.
Jacob Devlin [16] told that BERT is simple both conceptually and experimentally. It produces new state-of-the-art outcomes for eleven Natural Language Processing tasks. There are two approaches to using pre-trained language representations in downstream tasks: feature-based (like ELMo) and fine-tuning (such as open AI GPT). Pre-trained representations decrease the need for numerous highly developed task-specific architectures. BERT is the first representation model based on fine-tuning to outperform a broad range of task-specific architectures on a variety of sentence- and token-level tasks. BERT advances the state of the art in eleven NLP tasks.
Hu Xu [17], in new advancements in BERT-based ABSA language models, inspired this research. Because NSP is not commonly utilised for pre-trained models, we employ masked language model (MLM) instead of next sentence prediction (NSP) for fine-grained token-level features (Liu et al., 2019). The training corpus is built up of more than 20 GB of Amazon reviews (He and McAuley, 2016) and Yelp review datasets 2. We begin by fine-tuning BERT for four epochs on such a corpus. MLM learns highly fine-grained features, the majority of which are dedicated to domains and semantics of features instead of views, according to this research.
TABLE 1 gives an overview of the papers discussed.

IV. System Model

This section shows the flow chart of Sentiment Analysis process and it also comprises of BERT Architecture in depth.

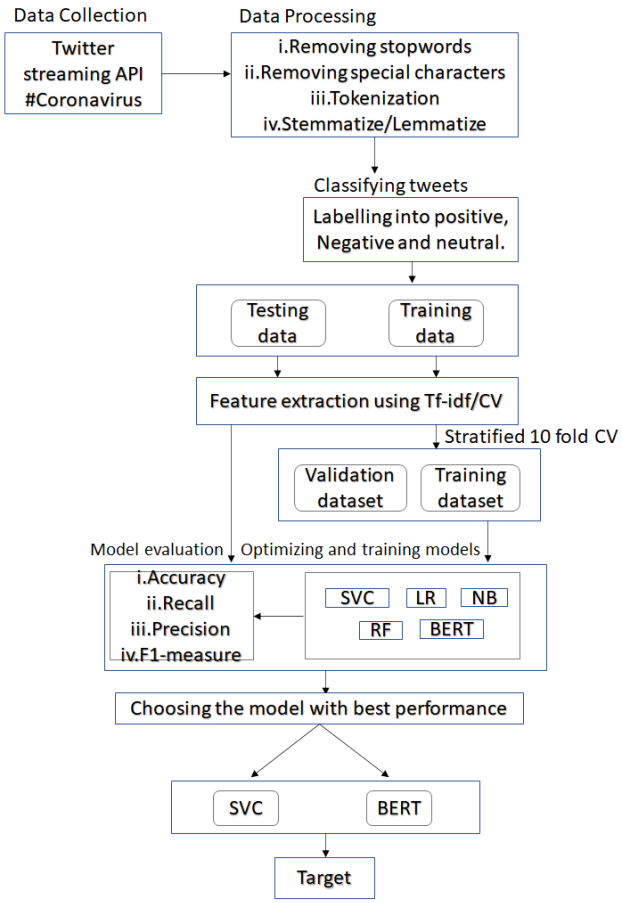

Fig. 1. **Flowchart of Sentiment analysis process**

Fig. 1. shows the flow of the procedure that we have done so far to choose model for the better performance.

A. BERT MODEL

BERT refers for Bidirectional Encoder Representations from Transformers, and it was created to condition all layers on both left and right context to pretrain deep bidirectional representations from unlabeled text data. Without significant task-specific architecture modifications, BERT is primarily employed for question answering and language inference.
The two steps in our methodology are pre-training and fine-tuning. During pre-training, the model is trained on unlabeled data across a variety of pre-training tasks. The BERT model is fine-tuned by first initialising it with pre-trained parameters and then fine-tuning all of the parameters with labelled data from downstream jobs. BERT comes in two sizes $BERT_{base}$ and $BERT_{large}$.

BERT also offers four types of pre-trained versions, depending on the scale of the model architecture. They are:
BERT-Base, Uncased: 12-layers, 768-hidden, 12-attention-heads, 110M parameters.
BERT-Large,Uncased: 24-layers, 1024-hidden, 16-attention-heads, 340M parameters.
BERT-Base, Cased: 12-layers, 768-hidden, 12-attention-heads , 110M parameters.
BERT-Large, Cased: 24-layers, 1024-hidden, 16-attention-heads, 340M parameters.
BERT relies on a Transformer (the attention mechanism that studies context-specific relationships between words in a text). A simple Transformer is made up of an encoder that interprets text input and a decoder that produces a task prediction. As the goal of BERT is to build a language representation model, it only requires the encoder part. The BERT encoder receives a string of tokens, which are then converted into vectors and analysed by the deep learning model(like Neural Network). Before BERT can begin processing, the info have to be massaged and improved with added meta - data: As mentioned below in Fig. 2., BERT comprises of ensuing input embeddings. They are: **Token embeddings**: In this embedding, a [CLS] token is added to the input word tokens at the beginning of the first sentence, and a [SEP] token is inserted at the end of each sentence.

Segment embeddings: Each token has a marker that indicates whether it is Sentence A or Sentence B. As a result, the encoder is able to distinguish between sentences.

Positional embeddings: Each token is assigned a positional embedding to indicate where it fits in the sentence.

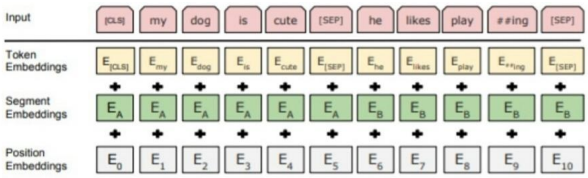

Fig. 2. **BERT Architecture**

We pre-train BERT using two unsupervised tasks:

Task 1. **Masked LM**: To prepare a deep bidirectional representation, we merely mask a percentage of the input tokens at random and then anticipate those masked tokens. This is referred as "masked language model(MLM)". Mask 15percent of the phrases in the input at random with a [MASK] token, then run the whole sequence through the BERT attention-based encoder, forecasting only the masked phrases utilising context offered by the non-masked phrases in the sequence. However, this naive masking strategy has a fault: the model only attempts to predict the appropriate tokens when the [MASK] token is visible in the input, whereas we need the model to attempt to forecast the correct tokens irrespective of which token is available in the input.

To address this problem, 80percent of the tokens chosen for masking are changed with the token [MASK].
Tokens are modified 10percent of the time with random tokens. Tokens are kept unaltered 10percent of the time.

Task 2. **Next Sentence Prediction (NSP)**: The BERT training procedure also use next sentence prediction to comprehend the relationship between two sentences. For activities like question answering, pre-trained model with this level of knowledge is useful. During training, the model is given pairs of sentences as input and is taught to predict whether the second sentence is also the upcoming sentence in the actual text.

As we saw before, BERT separates sentences using a specific [SEP] token. During training, the model is presented with two input sentences at a time:

50 percent of the time, the second statement comes after the first. Half of the time, it's a random sentence from the document.

The entire input sequence is processed by a Transformer-based model, the output of the [CLS] token is converted into a 2×1 shaped vector using a simple classification layer, and the Next-Label is allotted using softmax to determine whether the second phrase is related to the first.

Masked LM and Next Sentence Prediction were used to train the model. This is done to reduce the aggregate loss function of the two techniques – making them "better together".

In a range of general language comprehension tasks, including natural language inference, sentiment analysis, question answering, paraphrase recognition, and linguistic acceptability, BERT excelled the best. Because encoding a concatenated text pair using self-attention comprises bidirectional cross attention between two phrases, BERT instead employs the self-attention mechanism to integrate these two stages. Fine-tuning is considerably inexpensive as compared to pre-training. Starting with the identical pre-trained model, all of the research findings can be duplicated within an hour on a single Cloud TPU or a few hours on a GPU. For 11 NLP tasks, BERT gives fine-tuned results. Some of the outcomes on benchmark NLP tasks are discussed here.

GLUE: A number of Natural Language Understanding activities make up the General Language Understanding Evaluation task. The classification layer weights W (K×H), where K is the number of labels, were the only new parameters introduced during fine-tuning. Use a batch size of 32 for all GLUE projects, then fine-tune the data over three epochs. For each job on the dev set, we choose the best fine-tuning learning rate(from 5e-5, 4e-5, 3e-5, and 2e-5).

SQuAD v1.1: The Stanford Question Answer Dataset is a set of 100,000 Question Answer Pairs gathered from the public. The goal is to guess the length of the answer text from the passage. With a learning rate of 5e-5 and a batch size of 32, fine-tune for three epochs. The best performing BERT (with the ensemble and Trivia QA) exceeds the top leaderboard system by 1.5 F1-score in ensembling and 1.3 F1-score overall. In fact, in terms of F1-score, a single BERTBASE surpasses the best ensemble system.

SWAG: The Circumstances With Adversarial Generations dataset contains 113k sentence completion challenges that use rooted logical reasoning to decide which answer is the best fit. The goal is to pick the most logical continuation from a list of four possibilities fine-tune the model for 3 epochs at 2e-5 learning rate and 16 batch size. The OpenAI GPT is outperformed by $BERT_{large}$ by 8.3percent. It even exceeds a highly skilled person.

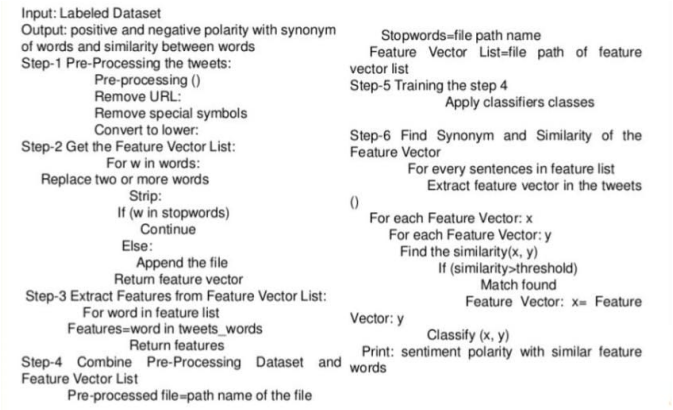

Fig. 3. **Pseudo code of the Process**

Fig 3 depicts a pseudo code that uses synonyms to identify positive and negative polarity.

V. Experimental Details

Libraries, Softwares, and Machines are all included in this section. We imported numerous libraries, including numpy, which is used for scientific computing in Python. Pandas, which is used for data manipulation, matplotlib, which contains bars, pies, lines, and scatter plots, and seaborn, which is used to create statistical graphics. We used Google collab to implement our source code as it is more compatible with python code execution.

A. Experimental Setup

COVID-19 tweets are used to construct datasets. The sentiments in these tweets have been stemmatized/lemmatized and grouped. These tweets are ingested into the train and test processes. Based on their sentiments, these tweets are categorised as 0,1,-1. We employ n-grams to extract features from a text corpus for machine learning algorithms. The sentence is broken into tokens, which are then passed to the ngrams function.

TABLE I
PREVIOUS SURVEY COMPARISON

PAPER RESOURCES				
Paper ID	Description	Methodology	Data sets	Conclusion
[2]	The goal of this study is to show how social media sentiments concerning corona virus.	Latent Dirichlet Allocation,lexicon based approach	tweets on Coronavirus produced by Christian	To undertake sentiment analysis of the gathered tweets,this study employs lexicon-based methodologies to characterise folk's feelings depending on the most widely discussed subjects.
[3]	This article examines the worldwide sentiment analysis of tweets about covid, as well as the changes in sentiment noticed over time.	Exploratory Data Analysis ML with Classification Methods LSTM and ANN	A.Coronavirus Tweets Dataset ,B.Online Learning Dataset (Tweets), C.Work from Home Dataset (Tweets)	The research took into account the positive and negative sentiments, fear, and trust emotions expressed in the tweets.
[1]	The goal of this research is to examine Indian netizens' tweets during the COVID-19 lockdown.	(BERT) Bidirectional Encoder Representations from Transformers model, LR, SVM, LSTM	github.com (https://github.com/gabriel preda/CoViD-19-tweets (accessed on 12 January 2021))	This article examines public perceptions of epidemics and offers advice to medical officials, the general public, and private sector workers on how to resist overreacting to pandemics.
[16]	The pre-trained BERT model can be fine-tuned with an extra output layer to build state-of-the-art models for a range of tasks.	pre-trained BERT, ELMo,Glue, SQuAD v1.1 and SQuAD v2.O	WordPiece embeddings	The role is to apply the insights to deep bidirectional architectures ,permitting the same pre-trained model to deal with a range of NLP activities successfully.
[17]	This study explores the pre-trained hidden representations for tasks gained from BERT reviews (ABSA)	masked language model in BERT	A mixture of Amazon and Yelp review datasets	Pretrained BERT is good for AE or the extraction part of E2E-ABSA in this paper, but not so good for summarising opinions in ASC or detecting polarity in E2E-ABSA.

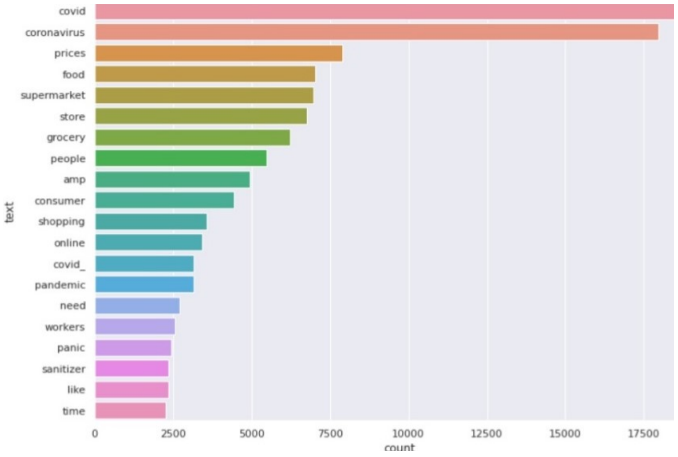

Fig. 4. **Unigram df**

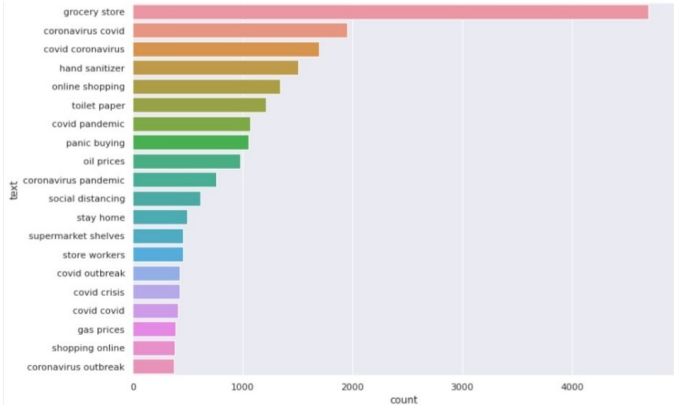

Fig. 5. **Bigram df**

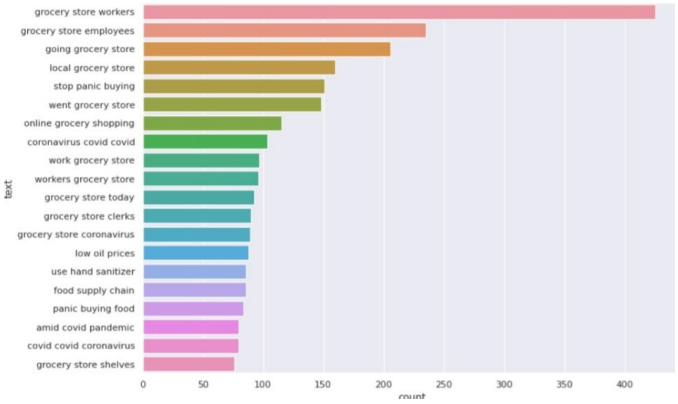

Fig. 6. **trigram df**

As expressed in Fig.4, Prices' being the most frequent unigram after covid/coronavirus may be due to rising food prices and other various shortages.

Grocery store way outpacing covid bigrams is pretty good. Online shopping, hand sanitizer, toilet paper, and panic buying are all within the realm of expectation as seen in Fig. 5.

Grocery store dominates these trigrams as shown in Fig. 6. People may be concerned about the safety of grocery shopping during a pandemic, and the health of the grocery store workers.

We will try different classifiers here: SVC, Logisitic Regression, Naive Bayes, and Random Forest, KNN, Decision Tree. Furthermore, we'll also be testing whether these models perform better using Term Frequency Inverse Document Frequency or just a simple Count for the vectors we feed into the model. TFIDF increases every time a word appears in a

document(tweet), but is then offset for every document(tweet) that word appears. This can help pick out the more important words for classification. Additionally, we'll be using cross validation to help gauge each model's accuracy and variance across multiple splits of the data. If we want better accuracy, we should try BERT. We'll fit a BERT model and see how well it does.

B. DataSet

We used an opensource of textual records to assess human emotions and concerns in regard to COVID-19. We extracted 41158 tweets from the 16th of March, 2020 to the 14th of April, 2020 in order to reach our target. A geolocation filter was used to identify the nations by which the tweets arises. These tweets were gathered from all over the world. We sketched some of the typical locations found in tweets, as shown in Fig. 7, to assist us in appropriately mapping additional tweets to a specific country. The vast majority of tweets originate in English-speaking countries, which makes logical given that all of the tweets are written in English. The United States is the biggest contributor, followed by the United Kingdom and Canada.

Fig. 7. **Number Of Tweets By Country**

The columns present in train and test dataset includes the Username, Screen Name, Date, Location, Original Tweet and Sentiment.

C. Results and Discussion

For the entire world, the number of instances, sentiment (positive, neutral and negative), fear, and trust emotions associated with COVID-19 tweets were plotted and examined. When compared to other algorithms, SVC and Logistic Regression performs better. The median of SVC with tfidf vectorizers and Logistic with count vectorizers is nearly identical, but SVC has less variation and a little more even distribution. With that in mind, we'd choose the LinearSVC with tfidf and no lemma/stem over the logistic regression because it takes MUCH less time to execute and has less volatility based on these results.

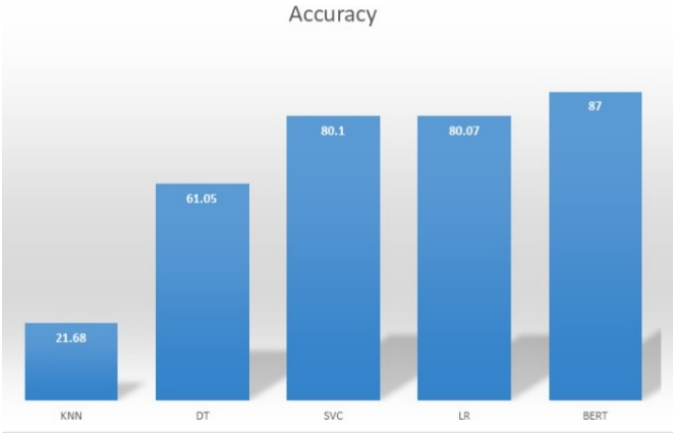

Fig. 8. **Graph shows accuracy of different algorithms**

An SVC employing tfidf vectors is recommended for efficiency. Let's put one on and look at the results more closely. The model achieves an 85 percent on the actual test data, which is 6 percent higher than the Linear SVC's performance. Additional hyperparameter adjustment, as shown in Fig. 8, may be able to squeeze out a little more accuracy from this BERT model.

VI. Conclusion

This paper's research involved sentiment analysis of covid tweets as well as establishing algorithm accuracy. From preprocessing until achieving maximum performance, the implementation was broken down into multiple parts. The produced datasets were analysed for positive, neutral, and negative attitudes, as well as anxiety and believe emotions expressed-Natural Language Processing: State of The Art, CurrentTrends and Challenge in the tweets. This study used the Google Collab platform and Python to improve the evaluation and processing of a huge number of relevant tweets. The plan is to label the produced dataset with its sentiment and employ other ML methods in the future, then compare the results of the labelled dataset with the results of the sentiment analysis accomplished in this study.

References

[1] Nalini Chintalapudi, Gopi Battineni and Francesco Amenta,"Sentimental Analysis of COVID-19 Tweets Using Deep Learning Models",1 April 2021, pp:329-339, doi: 10.3390/idr13020032.

[2] Muvazima Mansoor1, Kirthika Gurumurthy2, Anantharam R U3, and V R Badri Prasad,"Global Sentiment Analysis Of COVID-19 Tweets Over Time", October 2020.

[3] Manal Abdulaziz1, Mashail Alsolamy3, King Abdulaziz ,Alanoud Alotaibi2, Imam Mohammad Ibn Saud ,Abeer Alabbas4, "Topic based Sentiment Analysis for COVID-19 Tweets", Article in International Journal of Advanced Computer Science and Applications.published on: January 2021, vol. 12, pp:626-636, doi: 10.14569/IJACSA.2021.0120172.

[4] W. Y. Chong, B. Selvaretnam and L. Soon, "Natural Language Processing for Sentiment Analysis: An Exploratory Analysis on Tweets," 2014 4th International Conference on Artificial Intelligence with Ap-

plications in Engineering and Technology, 2014, pp. 212-217, doi: 10.1109/ICAIET.2014.43.

[5] Diksha Khurana1, Aditya Koli, Kiran Khatter, and Sukhdev Singh,"Natural Language Processing: State of The Art, Current Trends and Challenges", August 2017.

[6] Krishna Prakash Kalyanathaya, D. Akila and P. Rajesh,"Advances in Natural Language Processing –A Survey of Current Research Trends, Development Tools and Industry Applications",International Journal of Recent Technology andEngineering (IJRTE) February 2019, ISSN: 2277-3878, Volume-7, Issue-5C.

[7] Annina Simon1, Mahima Singh Deo,S. Venkatesan, D.R. Ramesh Babu,"An Overview of Machine Learning and its Applications",n International Journal of Electrical Sciences Engineering, Volume 1, Issue 1; 2015 pp. 22-24 @ Dayananda Sagar College of Engineering, Bengaluru-78.

[8] Girish, L. and Thara, D.K., 2015. Efficient virtual machine memory transfer in datacenter with optimal downtime. International Journal of Engineering Trends and Technology (IJETT), 23(9).

[9] Kozhevnikov, V.A., Pankratova, E.S., "Research of text pre-processing methods" for preparing data in Russian for machine learning", 2020.

[10] Sahana, D. S., and L. Girish. "Automatic drug reaction detection using sentimental analysis." International Journal of Advanced Research in Computer Engineering Technology (IJARCET) Volume 4, 2015.

[11] Wahab Khan, Ali Daud, Jamal A. Nasir, Tehmina Amjad, "survey on the state-of-the-art machine learning models in the context of NLP", pp:95-113, 2016A.

[12] Juan Ramos,"Using TF-IDF to Determine Word Relevance in Document Queries",Rutgers University, 23515 BPO Way, Piscataway, NJ, 08855, 2013.

[13] Anita Kumari Singh, Mogalla Shashi, "Vectorization of Text Documents for Identifying Unifiable News Articles",(IJACSA) International Journal of Advanced Computer Science and Applications, vol.10, No. 7, 2019, p.305, doi :10.14569/IJACSA.2019.0100742.

[14] Manish Munikar, Sushil Shakya and Aakash Shrestha,"Fine-grained Sentiment Classification using BERT", November 2019, doi: 10.1109/AITB48515.2019.8947435.

[15] Khanum, Salma, and L. Girish. "Meta heuristic approach for task scheduling in cloud datacenter for optimum performance." International Journal of Advanced Research in Computer Engineering Technology (IJARCET) 4.5 (2015): 2070-2074.

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova,"BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding",arXiv:1810.04805v2 [cs.CL] 24 May 2019.

[17] Hu Xu ,Lei Shu,Philip S. Yu and Bing Liu1, "Understanding Pre-trained BERT for Aspect-based Sentiment Analysis".Proceedings of the 28th International Conference on Computational Linguistics,(Barcelona, Spain (Online)), 2020, pp. 244–250, doi:10.18653/v1/2020.coling-main.21.

[18] Girish, L., and Sridhar KN Rao. "Anomaly detection in cloud environment using artificial intelligence techniques." Computing (2021): 1-14.

[19] Girish, L., and Sridhar KN Rao. "Quantifying sensitivity and performance degradation of virtual machines using machine learning." Journal of Computational and Theoretical Nanoscience 17.9-10 (2020): 4055-4060.