Media Engineering and Technology Faculty
German University in Cairo

Mapping and Tracking Sentiment Arcs in Social Media Streams

Bachelor Thesis

Author:	Maryam Ayman ElOraby
Supervisors:	Dr. Mervat Mustafa Abu ElKheir

Submission Date:	12 June, 2022

Media Engineering and Technology Faculty
German University in Cairo

Mapping and Tracking Sentiment Arcs in Social Media Streams

Bachelor Thesis

Author:	Maryam Ayman ElOraby
Supervisors:	Dr. Mervat Mustafa Abu ElKheir

Submission Date:	12 June, 2022

This is to certify that:

(i) the thesis comprises only my original work toward the Bachelor Degree

(ii) due acknowledgement has been made in the text to all other material used

Maryam Ayman ElOraby
12 June, 2022

Acknowledgments

This thesis would not have been completed without the support and kindness of the friends and family around me. To them, I give my heartfelt gratitude. I also wish to sincerely thank my supervisor, Dr. Mervat Abu ElKheir, for her guidance and valuable insights. I am extremely grateful for having worked with her.

Abstract

In recent years social media have emerged as a popular platform for people to share their thoughts and opinions on all kinds of topics and situations. Tracking sentiment and understanding a topic's evolution allows enterprises or governments to capture negative opinions and act promptly. We defined a sentiment analysis model architecture composed of BERT followed by a single-layer FFN with Softmax for multi-class classification. We tested it on TweetEval's benchmark for sentiment analysis, and the highest accuracy we achieved was 74.0%. We applied the model to time-stamped tweets concerning the COVID-19 pandemic - which caused quiet a buzz on social media - and plotted the temporal trends using a sliding window technique with Savitzky-Golay filter. We observed that the sentiment arc started at a low point at the beginning of the outbreak but gradually leaned towards the neutral. We compared the arc to that of vaccines and to the reported death rates; consulted the psychological research; and used the GSDMM topic modeling algorithm to explore and map events that might have influenced the resulting sentiment arc; in order to measure the arc's accuracy. The arc presented a degree of actuality, so we applied the model to tweets concerning Elon Musk's Twitter buyout deal to prove its ability to model sentiments toward other topics.

Contents

Chapter 1

Introduction

It is undeniable that social media platforms are omnipresent. Popular sites such as Twitter, Facebook, and Instagram have become a critical part of our everyday lives to stay connected, as well as to share and access a variety of content. Due to its international reach to populations at large, many organizations and individuals use social media to circulate their thoughts and perceptions.

This user-generated content is a bountiful source of user opinions and feedback, and mining it can benefit various applications that require an understanding of public opinion on a topic. For example, government organizations can benefit from observing public reactions and attitudes to various social issues and act accordingly. They could also gauge public opinion on policy announcements to better strategize and plan for the future [1]. Similarly, enterprises can capture negative or positive customer feedback about their products and use that information to improve the quality of their services.

However, emotions are not static but continuously evolve, unfold, fluctuate, and linger across time [2]. Thus, capturing the dynamic nature of human emotions through leveraging time series analyses would be advantageous in identifying patterns, regularities and trends; detecting abrupt shifts in sentiments and predicting people's future attitudes towards a topic [3].

1.1 Motivation and Objectives

According to Cambridge Dictionary[1], a sentiment refers to a general feeling, attitude, or opinion about something. Sentiments are typically classified into positive, negative, or neutral classes, and the process of computationally identifying the sentiment expressed in text is called sentiment analysis [4]. Finally, sentiment arcs are the plotting of the trajectories of sentiments as they change over time.

[1] https://dictionary.cambridge.org/dictionary/english/sentiment

The main objective of this thesis is to study and analyze sentiments revealed by people through their social media posts about a major event, and track how the sentiments change over time through the construction of a sentiment arc. We aim to create a sentiment analysis model with a high accuracy, to use for assigning sentiments to event-based time-series data. Then we plot them against time and analyze the temporal variations in sentiments, in attempt to map them the arc to topics or events. In the end, we prove that our model is generic, and could be extended to other domains.

1.2 Outline

This thesis consists of 5 chapters, including the “Introduction”. The content of each chapter is as follows:

- Chapter 2 “Background”: A brief background on techniques used in sentiment analysis. Related work on sentiment analysis and sentiment tracking is also discussed.
- Chapter 3 “Methodology”: A description of the proposed approach of this work.
- Chapter 4 “Experimentation & Results”: A description of the datasets used and the different experiments. A discussion of the results and the limitations faced in this project are also presented.
- Chapter 5 “Conclusion & Future work”: A summary of the thesis and suggestions for further work.

Chapter 2

Background

Sentiment analysis or opinion-mining is a sub-domain of natural language processing (NLP) and text analytics that focuses on discovering a writer's stance in written text. It has been applied in social media monitoring, tracking disasters, customer reviews, brand market monitoring and other applications

2.1 Social Media Analytics

Increasingly, social media is playing its role as a platform for individuals to express their wide range of views to a considerable number of people. While the number of social media users is continuously increasing, there is a demand to design techniques and tools to analyze such massive data [5]. Opinion mining is one example of the research conducted on social media text to take advantage of the large amount of online user-generated opinionated data.

There are many tools for tracking and measuring people's sentiments about any particular topic on social media, and Social Mention[1] is one of them. This social media monitoring tool aggregates user-generated content from different social media platforms.It allows the tracking of what people are saying about a product, a topic, a company or a person.

2.2 Sentiment Analysis

The typical approach is to identify a text's sentiment polarity as positive, negative or neutral, and it can be performed manually, semi-automatically, or automatically. The manual approach gets cumbersome as the corpus size increases [6]; it is labor-intensive, time-consuming, and thus usually used as a check on automated approaches. In literature,

[1] https://brandmentions.com/socialmention/

the automation of sentiment classification has been addressed mainly through dictionary-based or machine learning-based approaches. Some researchers have relied on a hybrid approach [7, 8].

The accuracy of classifying documents correctly as positive, neutral, or negative is commonly measured using accuracy, precision, recall, and F1-score [9].

2.2.1 Lexicon-Based Approach

The lexicon-based approach can be divided into dictionary-based and corpus-based approaches. The dictionary-based approach begins with finding sentiment or opinion seed words with known positive or negative orientation and then searches a dictionary, like WordNet [10], for their synonyms and antonyms.

The corpus-based approach depends on a predefined seed list of sentiment words and then finds other words using statistical or semantic methods in a large corpus to help find sentiment words with domain-specific orientations. This approach is used to adapt a general sentiment lexicon into a new domain-specific lexicon [11].

Bonta et al. [12] carried out a comparison between different lexicon-based libraries and tools like SentiWordNet, TextBlob, and VADER, to find the most effective one for sentiment analysis. Using a labeled set of 11,861 movie reviews that were extracted from a movie review-aggregation website, they observed that VADER, a lexicon-based method attuned specifically to social media texts such as tweets [13], performed the best with an accuracy of 77.0% and F1-score of 81.60%.

Sentiment lexicons provide knowledge on prior polarity (positive, negative, or neutral) of a word, i.e., its polarity in most contexts. However, when the complexity of human language is considered, the lexicon-based approach suffers from poor recognition of sentiments; it does not account for the context-dependence of the sentiment orientation or polarity. The word “good” usually indicates a positive sentiment as in “I had a good time” whereas it could imply a neutral sentiment as in “I want a good car”.

2.2.2 Machine Learning Approach

There are two main methods in machine learning: unsupervised and supervised learning. However, most existing research for document-level sentiment classification use the latter strategy.

In supervised learning, a labeled training set is required to train and structure the algorithm, which will then automatically predict the label of new data. However, selecting and extracting the appropriate set of features is critical to the model’s success before training, as it extracts valuable information from the data. Feature extraction is crucial in information retrieval (IR) and natural language processing, as machine learning algorithms cannot directly work on the raw text. So, we need to convert text into a numerical representation.

Bag-of-Words

The bag-of-words (BoW) model is a simplified text representation technique where, given a collection of documents, the set of words used in the entire collection is first identified. Commonly called vocabulary, this set is traditionally reduced by keeping only the words used in at least two documents or by removing stop words such as “and”, “because”, “to”, “of”, or “you”.

Following that, each document can be represented as a vector of length equal to the vocabulary size. The vector could either be a binary representation of the words present in the document, or a vector of word counts.

The BoW model has, nonetheless, some limitations despite being simple and easy to implement: the word order is lost, causing that different sentences may have the same representation when the same words are used. Intending to reduce the impact of this limitation, a particular variant of the BoW technique corresponds to N-grams, which identifies multi-word expressions occurring in the document, ensuring word order in short contexts. Expressions like “a year earlier” and “United States” will be detected as single units when using a tri-grams or a bi-grams model, respectively. Still, the BoW representation and its variants have little sense about the semantics and context of the words. This yields low accuracy in tough and challenging classification problems as sentiment analysis, where the sentence should be seen and understood as a whole. BoW representation also suffers from sparsity and high-dimensionality problems, as it uses a large memory space. For example, in a case where the vocabulary size is 10,000, a sentence with 5 words will need a vector of 10,000 tokens. With 20,000 tuples (sentences) in the document, the document will be represented by 20,000 vectors, with a size of 10,000 tokens each.

TF-IDF

Another document representation approach is the vector space model, commonly used to compute a degree of similarity between documents where each document is represented as a vector of feature weights. The feature weights can be computed in several ways, including the TF-IDF weighting scheme.
The idea behind it is that some words frequently occur in many documents but they are not necessarily important, like stop-words. The TF-IDF weighting scheme combines the individual frequency for each term or feature f in the document d (i.e., $TF_{f,d}$) with the inverse frequency of the feature f in the entire collection of documents (i.e., IDF_f).
There are different ways to compute the term frequency (TF). However, the most common way is counting the number of occurrences of a feature f within a document d.
The inverse document frequency (IDF) measures feature importance within the corpus. It is the inverse of the number of documents in which a feature occurs, and is computed as follows:

$$\mathrm{IDF}_f = log\left(\frac{N}{d_f}\right) \tag{2.1}$$

where N corresponds to the number of documents in corpus and d_f corresponds to the number of documents containing a feature f.
Term Frequency - Inverse Document Frequency (TF-IDF) combines both as follows:

$$\text{TF-IDF}_{f,d} = \text{TF}_{f,d} \times \text{IDF}_f \tag{2.2}$$

Word Embeddings

Another type of text representation is word embeddings. To begin, embeddings are low-dimensional representations of points in a higher-dimensional vector space. In the same manner, word embeddings are dense vector representations of words in lower-dimensional space. The advantage of this approach is that it tends to capture the semantic meaning of words in text; It assigns similar vector representations to words with similar meanings.

One example is the Word2Vec algorithm that uses a neural network model, where the neural network learns word associations or dependencies from a large corpus [14]. Word2Vec's embeddings are key-value pairs, essentially a 1-1 mapping between a word and its respective vector; Word2Vec takes a single word as input and outputs a single vector representation of that word. Word2Vec finds similarities among words by using the cosine similarity metric. If the cosine angle is 1, that means words are overlapping. If the angle is 90, words are independent or hold no contextual similarity. In theory, when using the Word2Vec model, the pre-determined Word2Vec embeddings are only needed.

Global Vectors for Word Representation (GloVe) [15] is another word embeddings algorithm developed by researchers at Stanford University. It aimed to extend Word2Vec to capture global contextual information in a corpus; Word2Vec only considers neighboring words during training. GloVe, on the other hand, considers the entire corpus and creates a large matrix that captures the co-occurrence of words within the corpus. However, the co-occurrence is still defined upon local context windows, so GloVe basically captures local context similarity of words as well [16].

Bidirectional Encoder Representations from Transformers

Introduced by Google in 2018, Bidirectional Encoder Representations from Transformers (BERT) is a massive pre-trained, bidirectional encoder-based transformer model that comes in two sizes: $\text{BERT}_{\text{Base}}$ which has 12 encoder blocks, 12 attention heads, and 110 million parameters; and $\text{BERT}_{\text{Large}}$ which has 24 encoder transformer blocks, 16 attention heads, and 340 million parameters.

BERT relies on the attention mechanism for generating word embeddings to generate high-quality context-aware embeddings that allow for multiple representations. Embeddings are refined during training by passing through each BERT encoder layer. For each word, the attention mechanism simultaneously learns word associations based on the words on the left and the right. Accordingly, BERT takes as input a sequence rather than a single word. Positional encoding is also applied to keep track of the positions of words in a sentence, while the previously mentioned models do not account for word positions. With BERT, the actual model is needed as the vector representations of words will vary based on the specific input sequences. The output is a fixed-length vector representation of the input sentence.

BERT is more advanced than any of the techniques discussed before; it creates better word embeddings as the model is pre-trained on 3.3 Billion words including the entire Wikipedia. BERT was described by Google as "one of the biggest leaps forward in the history of Search" [17] as they started using it to enhance their search engine. This was proven by the results published in their released paper showing that BERT had achieved more profound contextual representations than previous language models.

BERT is extremely versatile as it can be used for many tasks, including translation, question answering, sentiment analysis, and text summarization. This is achieved by fine-tuning the embeddings on task-specific datasets.

Machine Learning Algorithms

There are several machine learning classification algorithms, and Decision Tree is one of them. It is structured as a tree, where the internal nodes represent the features, outcomes are associated with each leaf node, and the branches represent the decision rules.

The Random Forest is an ensemble of decision trees, where each tree is trained independently. For classification tasks, the output of the Random Forest is the class selected by most trees.

Another classifier is the XGBoost which is based on a gradient boosting algorithm. Its main idea is to learn from the previous errors performed by the model while training.

The Naïve Bayes algorithm is a classification algorithm based on Bayes rule and a set of conditional independence assumptions. It has several variations, including the Multinomial Naïve Bayes, Bernoulli Naïve Bayes, and Gaussian Naïve Bayes.

Logistic Regression is another machine learning algorithm that uses the logistic function $\frac{1}{1+e^{-x}}$ to model a binary output variable [18]. However, it can be extended into multiple classes (then it is called Multinomial Logistic Regression)

A Support Vector Machine (SVM) is a supervised machine learning model that uses classification algorithms and performs very well with a limited amount of data to analyze.

A comparative study classifying airline reviews reported that support vector machines and logistic regression multi-classification algorithms, with Bag of words (BoW) as features, provided 77% accuracy on a large, imbalanced, real word dataset, compared to Multinomial Naïve Bayes and Random Forest [19]. Using multinomial logistic regression, [20] observed that this method can accurately predict the sentiment of Twitter users up

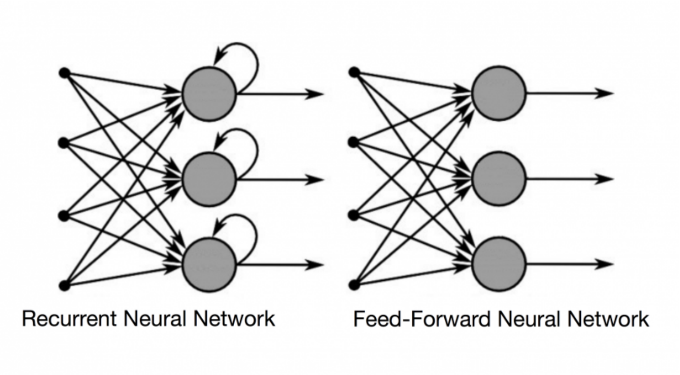


Figure 2.1: The comparison between Recurrent Neural Network (RNN) and Feed-Forward Neural Network (FNN). It demonstrates in FNN there is only one direction for the data to move, whereas in RNN there is a loop. *Adapted from: [23].*

to 74% on a manually labelled dataset with a composition of 90% training data and 10% testing data.

The need for labeled data, features extraction from text data, and selecting a suitable classification algorithm for the tasks deem the traditional machine learning algorithms complex and lacking portability or adaptability across different domains.

Neural Networks

Deep learning has emerged as a powerful machine learning technique in the last decade, producing state-of-the-art results in a wide range of application domains, including computer vision, speech recognition, and natural language processing (NLP) [21]. Deep learning for sentiment analysis has recently gained considerable popularity [22].

Inspired by the structure and function of the human brain, a neural network can learn to correct itself when it makes an error. This is possible through the neural network architectures that consist of many information processing units (called artificial neurons) organized in layers. It learns to perform tasks by adjusting the connection weights between neurons. A neural network with multiple hidden layers is a deep neural network (DNN).

One of the commonly used deep learning models for NLP is the recurrent Neural Network (RNN). In a Feed-forward neural network, a traditional neural network, the information only moves in one direction: from the input layer, through the hidden layers, then to the output layer. The output is only dependent on the current input and has no memory of the previous input data. The RNN, on the other hand, has information cycling through a loop. It makes decisions based on current and previously received inputs, as shown in Figure 2.1.

Nevertheless, an RNN has its limitations when learning long-term dependencies; as the RNN processes more steps, it has trouble retaining information from previous steps due to the vanishing and exploding gradients problem [24]. Long short-term memory (LSTM) networks have thus been explored. LSTM networks have better capabilities for learning long-term dependencies using memory cells that are in charge of storing the information and "gates" that are adjusted by the network so it can remember what it needs and forget what is no longer helpful. Several LSTM architectures were later developed, including Bi-LSTM, which processes sequences in both directions, Stacked LSTM, and CNN-LSTM [25].

Transformers

RNNs are slow to train; they can not parallelize because they require words to be processed sequentially; hence we need inputs of the previous state to make any operation on the current state. In terms of learning the context of words, even bi-directional LSTMs (Bi-LSTM) learn left to right and right to left context separately and then concatenate them. So the actual context is slightly lost [26].

Transformers were introduced in 2017 by Google, as deep learning models in which each output element (word) is connected to every input element in a sentence, and the weightings between them are calculated dynamically based on their connection. In NLP, this mechanism is called attention [27]. Transformers are also designed to handle sequential input data. But unlike RNNs, they do not require the processing to be done in a fixed order, allowing for parallelization and thus reduction in the training time. Furthermore, context of words is better learned as they can learn from both directions simultaneously.

The transformer consists of two key components: an encoder and a decoder - each is a stack that can be called "transformer blocks". The encoder takes the input words simultaneously and generates embeddings for every word. These embeddings are vectors that encapsulate the meaning of words, and learns context. In contrast to the older context-free approaches to word embeddings like Word2Vec or GloVe, this takes into account neighboring words when generates a contextual representation for each word.
The decoder takes the embeddings from the encoder and produces a prediction for the task, and learns how words relate to each other.

Both of these components can be used independently and build systems that understand language. The stack of decoders is - in simple terms - the architecture of GPT-2 transformer model (OpenAI transformer) [28]. On the other hand, a stack of transformer encoder blocks gives BERT [26].

2.3 Sentiment Tracking

Reagan et al. [29] defined *emotion arcs* as the plotting or tracking of sentiment valence of some text form along the time axis. Their work presented an analysis of fictional texts. They discovered that only a few universal plot structures are linked to the evolution of the happiness emotion in stories over time. To create the arcs, they used a tool called

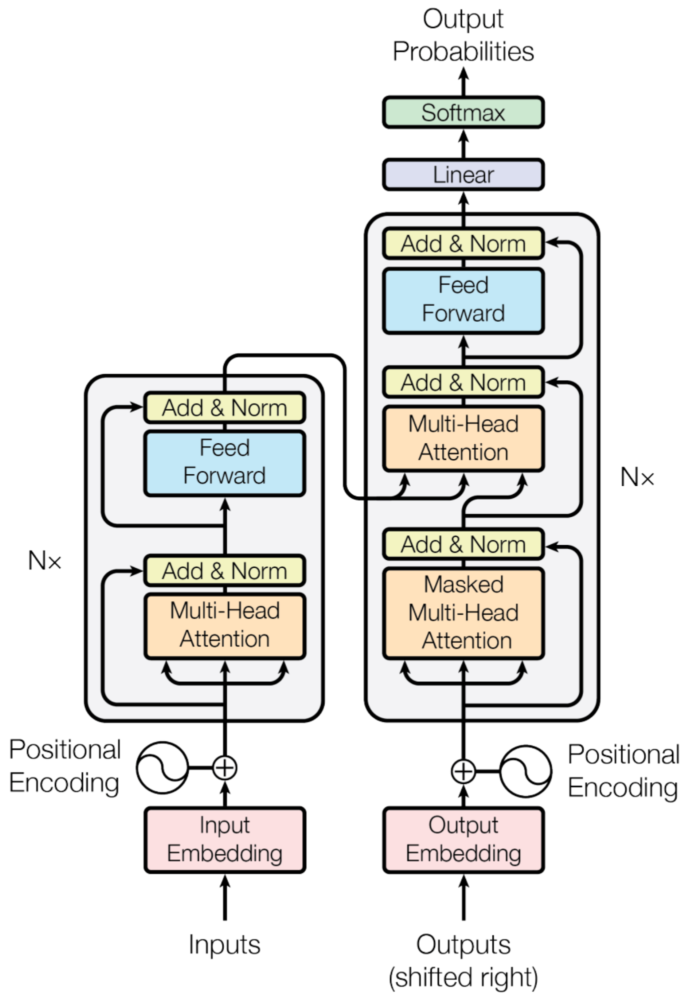


Figure 2.2: The transformer model architecture *Adapted from: [27].*

the Hedonometer[2] to score the sentiment of individual words, and then slid a window of size 10,000 words through the text and plotted the results against time. That made it possible to see how the mood changes throughout the text, revealing a kind of emotional narrative.

The Hedonometer was created by a group of researchers at the Vermont Complex Systems Center [30]. It has has been tracking people's happiness on Twitter by rating positive and negative words since 2008. To calculate the happiness in a text, the researchers first determined how happy individual words are. They asked 50 people to score about 10,000 words for happiness on a scale from 1 to 9 and then averaged the scores to produce a final value. The word "laughter" is the happiest, earning a score of 8.5. "terrorist" received the lowest score of 1.3. The scores were released in a dataset under the name of "LabMT".

[2] Hedonometer.org

Tourism attracts sentiment analysis researchers to monitor the decline in this sector and suggest recommendations for decision-makers. Thus, some researchers were interested in tourists' opinions of some touristic destinations [31]. They analyzed tweets about some destinations and plotted how people's collective sentiments changed over time. To create the sentiment analysis model, they collected hotel reviews along with their numeric rating, and a Naïve Bayes classifier was trained on this data to classify tweets into five classes: 1 meaning very negative, 2 meaning moderately negative, 3 denoting neutral, 4 meaning positive, and 5 meaning very positive. They used the model to annotate tweets revolving around a specific destination and plotted the scores against time. Figure 2.3 shows a resulted graph where the sentiment towards the word "Sri Lank" was tracked and observed over the period from October 2009 to September 2011.

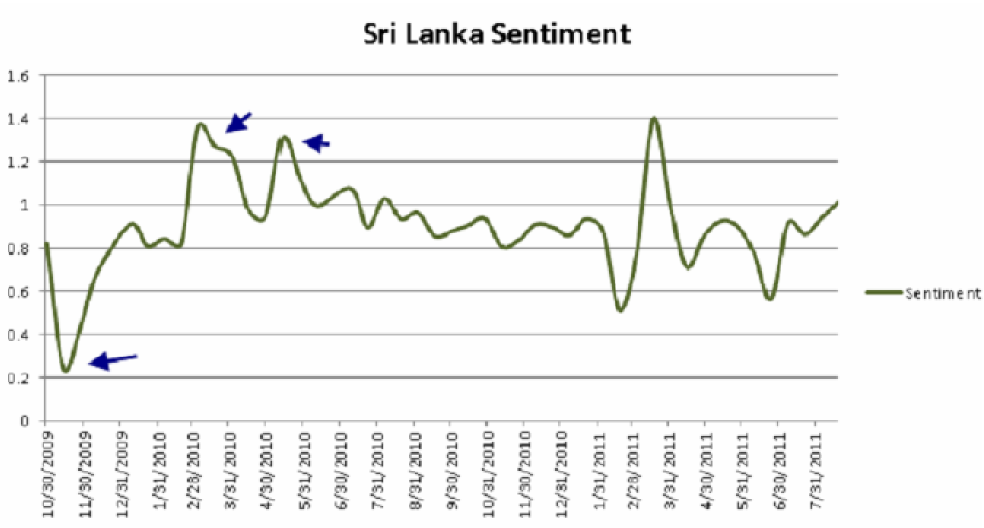


Figure 2.3: Sentiments towards "Sri Lanka" over time. *Adapted from: [31]*

When surveying the literature, we find numerous studies performing sentiment analysis in the health sector worldwide. Some have their datasets openly available. For example, *Lamsal, R.* [32] released an open-access billion-scale COVID-19 Tweets dataset that dates back to October 19, 2019, and is updated daily to date. They used TextBlob to estimate the sentimental polarity of words and tweets and classify them into positive, negative, and neutral classes.

Pone et al., [33] aimed to perform sentiment analysis on COVID-19 tweets, utilizing Lamsal's dataset. However, they claimed that the original scores supplied with the dataset were inaccurate due to the presence of noise in tweets, so they pre-processed the tweets by removing stop words, usernames, links, punctuation, and numeric values, followed by applying stemming techniques. They reapplied TextBlob on the cleaned tweets, then supported their theory using a manual label comparison and a performance comparison. The dataset was split into a training set (80%) and a testing set (20%), and different approaches for feature extractions were used: Bag of Words (BoW), Term frequency-inverse document frequency (TF-IDF), and a concatenation of both methods. They compared the performance of various ML models like Random Forest (RF), XGBoost, Support Vector Classifier (SVC), Enhanced Tree Classifier (ETC), and Decision Tree (DT). The performance was evaluated on accuracy score, precision score, recall score, and F1 score. They further evaluated the performance of their proposed model against VADER and an LSTM model with a dense layer and a dropout layer with a 0.5 dropout rate and against Bi-LSTM. Nevertheless, the study concluded that ETC is the best performer with the features concatenation approach they proposed. They suggested that the poor performance of the Deep learning model is due to the dataset's small size. This is in line with *Feng et al.* [34] who found deep learning models to perform poorly on small datasets.

In the political sector, *Wang et al.,* [1] created a system for real-time analysis of public sentiment toward presidential candidates in the 2012 U.S. election from tweets. They trained a Naive Bayes classifier on a dataset annotated by a crowd-sourcing approach to do sentiment annotation. Their model performed at 59% accuracy on the four-category classification of negative, positive, neutral, or unsure. They applied their classifier to a live stream of tweets. They plotted the results to a bar graph of the number of positive and negative tweets about each candidate in the last five minutes to indicate sentiment towards the candidates. Moreover, they used TF-IDF to find the most prominent words in the last five minutes.

In a more recent study aiming to find whether COVID-19 pandemic-related tweets were correlated with the overall election results during the period leading up to the United States 2020 election day, *Doman et el.,* [35] collected tweets related to COVID-19 pandemic from March 20, 2020, till November 4, 2020. Data on popular voting, organized by state, was obtained from a public, nonpartisan online source.

Sentiment analysis was performed using VADER after excluding retweets and duplicate tweets. The daily average sentiment intensity was computed, and a 14-day sliding time window was applied for time series analysis. In order to compare the tweet sentiment intensity between the two states, they estimated a ratio between the average sentiment intensity for the first state versus the second one in the sliding time windows.

They plotted the change of relative sentiment intensities of COVID-19 tweets in each state separately over time, and then the average ratio was estimated and plotted. Their results showed a weak correlation between the average sentiment of COVID-19-related tweets and popular votes in the 2020 election in the United States. However, they observed a shifting trend at the start of the pandemic: the overall sentiments in one state started gradually shifting from negative to positive. In contrast, an opposite trend was observed in the other state.

In general, we notice that Twitter is frequently used to analyze public emotion and the impact of major events or shocks around the world. For example, data extracted from tweets on investors' sentiments have been used to predict stock market movements in developed and emerging markets [36]. Multiple researchers have used the sentiment analysis of Twitter data to monitor short-run levels of happiness and life satisfaction [37][38] [39].

2.4 Trend Analysis

There were many other attempts to explain the sentiment trends and outliers, or the sudden shifts in sentiments (spikes). *Giachanou et al.,* [40] focused on the problem of tracking sentiment towards different entities, detecting sentiment spikes, and extracting and ranking the causes of a spike. Their approach leverages Latent Dirichlet Allocation (LDA) topic modeling algorithm for extracting the topics discussed in the time window before a sentiment spike and Relative Entropy to rank the detected topics based on their contribution to the sentiment spike.

Jang et al., [41] conducted a specialized study to monitor the COVID-19 debates and discovered that people showed negative sentiments towards the outbreak as a whole but had positive sentiments related to physical distancing. This indicated that different topics were likely to be associated with varying sentiment levels, and that undertaking sentiment analysis without combining the results with topic modeling would yield little. Topic Modeling is a technique to extract the hidden topics from a body of text.

Chapter 3

Methodology

Among the various social media platforms, Twitter is a gold mine with 500 million tweets per day from 330 million users [1] addressing topics across various domains (e.g., commerce, health, or disaster management). Thus, two main arguments for pursuing the massive data stream of Twitter for the initial experiment are: the current and growing importance of Twitter[2]; and the potential for describing universal human patterns, whether emotional, social, or otherwise.

Twitter stays on top for gathering news, particularly health-related news [42]. This was evident during the lockdown from COVID-19 in 2020, when people took to Twitter to express their viewpoints and feelings. The pandemic has caused a substantial spike in the number of daily tweets on Twitter, which has risen from just over 320 million messages per day in November 2018 to the highest daily tweet volume level of 500 million tweets and the rise has remained consistent for nearly three months [43]. The pandemic proposed an ideal opportunity to analyze public sentiment and its dynamics; thus, we choose it to be the initial focus of our analysis.
We later prove that architecture and method presented are generic, and thus, can be easily adopted and extended to other domains.

The proposed method begins with fetching a series of tweets related to the topic from Twitter, then pre-processing the data for a more accurate assessment by removing noise and redundant information. Later, the cleaned data is passed through a classification model constructed by fine-tuning BERT on a specific dataset of labeled tweets to predict the sentiments. We use the generalization given by *Liu* [44], where our goal will be to detect the average sentiment of a document using the words contained within. The results are then plotted against time to generate a sentiment arc.
GSDMM topic modeling algorithm is also applied, after further pre-processing, to extract other discussed topics from the data. The architecture of the proposed method for tracking sentiments is presented in Figure 3.1.

[1] Twitter Statistics. https://www.omnicoreagency.com/twitter-statistics/

[2] "Library of Congress will save tweets" New York Times. http://www.nytimes.com/2010/04/15/technology/15twitter.html

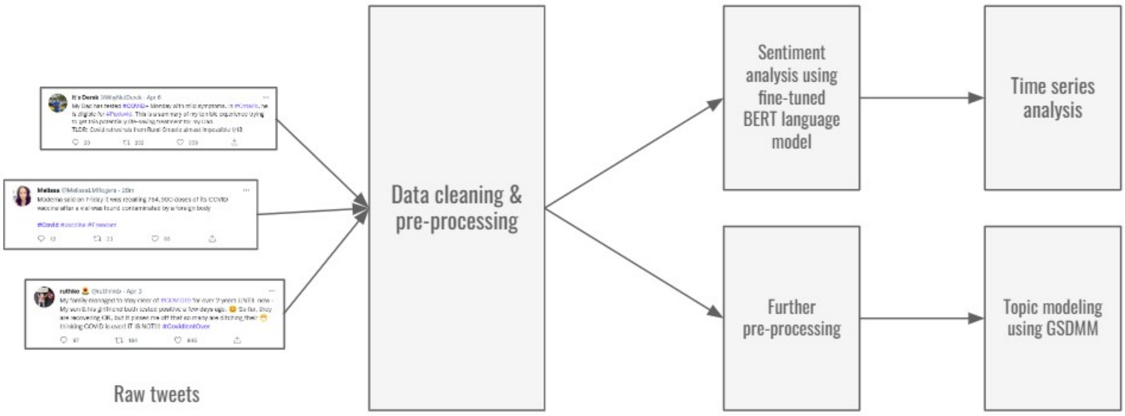

Raw tweets
Data cleaning & pre-processing
Sentiment analysis using fine-tuned BERT language model
Time series analysis
Further pre-processing
Topic modeling using GSDMM

Figure 3.1: Overview of the proposed tracking system architecture.

3.1 Sentiment Analysis Model

For the sentiment analysis model, we chose to fine-tune BERT for its ability to better understand the language. Therefore, this section covers some basics about BERT input representations and fine-tuning it.

3.1.1 Input Representation

BERT expects input sequence in a specific format: having a maximum size of 512, with the first token of the sequence being [CLS], and [SEP] special token for separating sentences. The final hidden state corresponding to the [CLS] token is used as the aggregate sequence representation for classification tasks [26]. Since BERT works with fixed-length sequences, and due to the variation in lengths of sentences in a dataset, padding or truncation is required to have all sentences with a single fixed length. This is satisfied with adding a special token [PAD].

Tokenization is performed using WordPiece embedding at subword level [45]. Thus, rather than just labeling out-of-vocabulary (OOV) words to catch all tokens, words that are unknown are decomposed into subword and character tokens. This retains some of the contextual meaning of the original word. For example, if "walk" and "##ing" are present in the vocabulary but "walking" is an OOV words, then it will be decomposed into "walk + ##ing". Subwords are then passed through a Token Embeddings layer to convert each token into a 768-dimensional vector representation.

The vectors are then passed through a segment embedding layer. For tasks that require a pair of input texts, this layer helps BERT distinguish between the two sentences by adding a learned embedding to every token indicating whether it belongs to sentence A or sentence B. Positional embedding is added to each token to indicate its position in the sequence. If an input consists of only one input sentence, its segment embedding will just be the a vector corresponding to the first sentence. A visualization of this construction is shown in Figure 3.2.

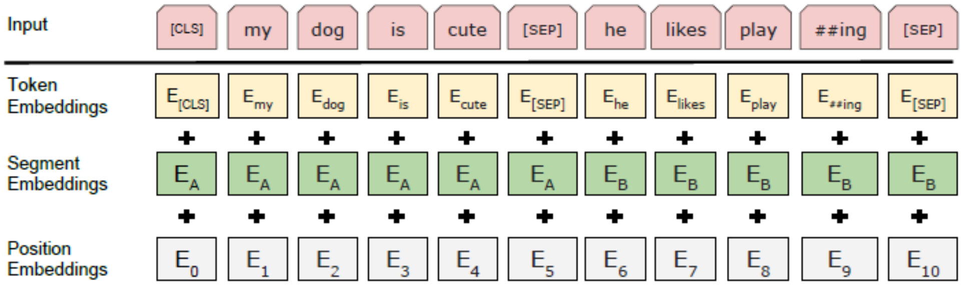

Input
[CLS]
my
dog
is
cute
[SEP]
he
likes
play
##ing
[SEP]
Token Embeddings
$E_{[CLS]}$ E_{my} E_{dog} E_{is} E_{cute} $E_{[SEP]}$ E_{he} E_{likes} E_{play} $E_{\#\#ing}$ $E_{[SEP]}$
Segment Embeddings
E_A E_A E_A E_A E_A E_A E_B E_B E_B E_B E_B
Position Embeddings
E_0 E_1 E_2 E_3 E_4 E_5 E_6 E_7 E_8 E_9 E_{10}

Figure 3.2: BERT input representation. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings. *Adapted from:* [26]

3.1.2 Pre-training

The versatility of BERT was achieved through a two-step process that begins with "pre-training". BERT learns language by simultaneously training on two unsupervised tasks: Masked language modeling (MLM) and next sentence prediction (NSP). MLM enables the bidirectional context learning from text by masking (hiding) a word in a sentence and forcing BERT to use the words on either side of the covered word to predict it. Whereas, NSP is used to help BERT learn about relationships between sentences by predicting if a given sentence follows the previous sentence or not.

The Google team open-sourced the model's code and made available for download versions of the model that had been pre-trained on massive datasets, namely, the entire English Wikipedia (2,500 million words), and the Book Corpus (800 million words). Of the different available versions, we have chosen to work with BERT$_{\text{BASE}}$ pre-trained model. In particular, we chose BERT$_{\text{BASE-CASED}}$ to preserve text case.

3.1.3 Fine-tuning

Following pre-training, BERT can be used with two approaches: one is with "feature extraction" mechanism. That is, we feed the final output of pre-trained BERT as an input to another model while preserving the architecture of the BERT model. This way, we extract features to use as contextualized word embeddings and then use them as an input to a separate model for the actual task at hand.

The other way is "fine-tuning" BERT. In this method, we modify the architecture by adding additional layer(s) on top of BERT and then train the whole thing together. This way, we train the additional layer(s) from scratch, but the BERT weights are only fine-tuned, so training time is fast.
Fine-tuning process teaches the pre-trained model to learn representations that are more useful to a specific task [46].

As shown in Table 3.1 adapted from [26], fine-tuning approaches achieve better performance results than feature-based approaches, where different combinations of hidden

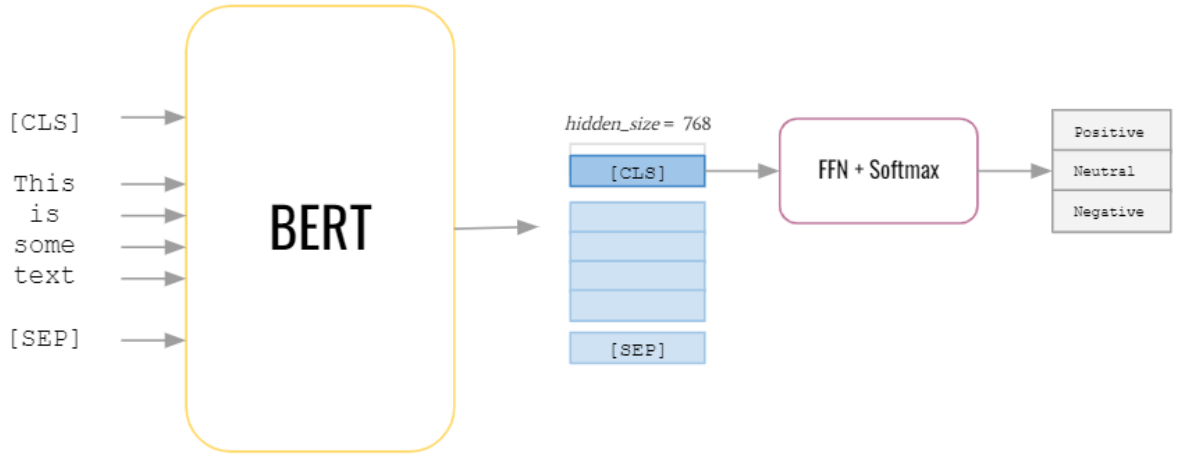


Figure 3.3: Overview of the sentiment analysis model architecture used in this thesis: The [CLS] Token output is used for classification along with an added FFN and Softmax layers.

vectors are experimented with. Therefore, in this thesis, we decide to use the fine-tuning approach with BERT for the sentiment analysis task.

BERT outputs vectors of size *hidden_size* (768 in BERT$_{\text{BASE}}$) for each input token in a sequence, starting with [CLS] and separated by [SEP]. BERT takes the final hidden state **h** of the [CLS] token as the representation of the whole sequence.

According to *Devlin et al.*, the authors of BERT paper, fine-tuning for classification tasks can be achieved by adding only one output layer, so a minimal number of parameters need to be learned [26]. Thus, in aim to meet the goal of this thesis, we add a simple single-hidden-layer feed-forward neural network, with softmax to the top of BERT to estimate the probability of a label c as:

$$p(c|\mathbf{h}) = \text{softmax}(W\mathbf{h}) \tag{3.1}$$

where W denotes the task-specific parameter matrix, and **h** denotes the final hidden state of [CLS] token. Fig 3.3 shows a visualization of this architecture.

The softmax function returns an estimated probability for the three classes, and the sum of these probabilities adds up to 1. To unify the range and prepare for sentiment tracking, we use the probabilities as sentiment scores or intensities, and scale them to cover a range from -1 (very negative) to +1 (very positive). To elaborate, samples estimated to belong to the negative class will have their confidence scores multiplied by -1, to cover a range from -1 to 0. while positive samples' scores are left as is to cover the range from 0 to +1. Neutral sentiments' scores are mapped to 0.

Table 3.1: Performance comparison of feature-based and fine-tuning approaches with BERT on CoNLL-2003 NER. *Adapted from: [47]*

System	Dev F1	Test F1
Fine-tuning approach		
$\text{BERT}_{\text{LARGE}}$	96.6	92.8
$\text{BERT}_{\text{BASE}}$	96.4	92.4
Feauture-based approach ($\text{BERT}_{\text{BASE}}$)		
Embeddings	91.0	-
Second-to-Last Hidden	95.6	-
Last Hidden	94.9	-
Weighted Sum Last Four Hidden	95.9	-
Contact Last Four Hidden	96.1	-
Weighted Sum All 12 Layers	95.5	-

3.2 Time Series Analysis

We apply the BERT-based sentiment analysis model to the a stream of tweets addressing the topic in concern, and plot the daily average against time. Different smoothing methods are also applied to filter the noise, and estimate the overall trend. The two filters considered are the moving average and Savitzky-Golay (savgol) filters.

The moving average filter is the most straightforward to understand and use. It works by recursively averaging a number of points. The Saviztky-Golay filter uses least squares polynomial fitting to fit data in the time domain across a moving window. The Saviztky-Golay filtering method is superior to averaging because it preserves data features such as peak height and width, which are typically flattened when using a moving average filter [48].

3.3 Topic Modeling

Analyzing the trends allows for identifying sudden sentiment changes and, more importantly, getting insights on what might have caused a spike. One way is through the investigation of the other discussed topics through the application of a topic modeling algorithm to extract topics, or through the use of word clouds to highlight popular words and phrases at that time.

The most popular text clustering algorithm is Latent Dirichlet Allocation (LDA). However, there are other approaches for different purposes, like Gibbs Sampling Dirichlet Multinomial Mixture (GSDMM) for shorter texts [49].

The main difference between the two is that LDA assumes that a document is a mixture of multiple topics and calculates each topic's contribution to the document. On the other hand, GSDMM is designed to detect topics in smaller documents and assumes the presence of only one topic per document.In view of the fact that short text contains few words, it appears more sensible to assume that each such document would contain a single topic. Thus, some argue that GSDMM is better suited for topic modeling in the case of short text [50].

Chapter 4

Experimentation & Results

4.1 Experiments Setup

Data curation has been performed on a 2.80 GHz personal computer with 16 GB of RAM. All other experiments have been performed on Google Colab[1] notebooks in NVIDIA Tesla T4 GPU initialized with a High-RAM of 25.46 GB.

4.1.1 Datasets

Covid-19 Twitter chatter dataset for scientific use

"Covid-19 Twitter chatter dataset for scientific use" [51] is an ongoing project dating back to January 2020, where researchers in Georgia State University's Panacea Lab are collecting COVID-19 tweets from the publicly available Twitter stream. They have made the dataset publicly available for scientific use and released two versions: a full one containing both tweets and retweets and a clean version containing no retweets. The tweets are collected using Twitter's trending topics and selected keywords. Keywords "coronavirus" and "2019nCoV" were used to collect tweets from January 1st to March 11th; COVD19, CoronavirusPandemic, COVID-19, 2019nCoV, CoronaOutbreak, coronavirus, WuhanVirus, covid19, coronaviruspandemic, covid-19, 2019ncov, coronaoutbreak, and wuhanvirus were later used to collect the rest of the data.
A visualization (Figure 4.1) of location-tagged tweets and their counts shared by the researchers showed that the majority of the location-tagged tweets were from US, India, and UK, and Brazil.

As per Twitter's Terms of Service, which do not allow the full JSON for datasets of tweets to be distributed to third parties, the dataset contained only tweet identifiers(IDs). In this thesis, we worked with the clean version and, due to the limited computation power at hand, we worked with a subset of 100,000 tweets per month.

[1] https://research.google.com/colaboratory/

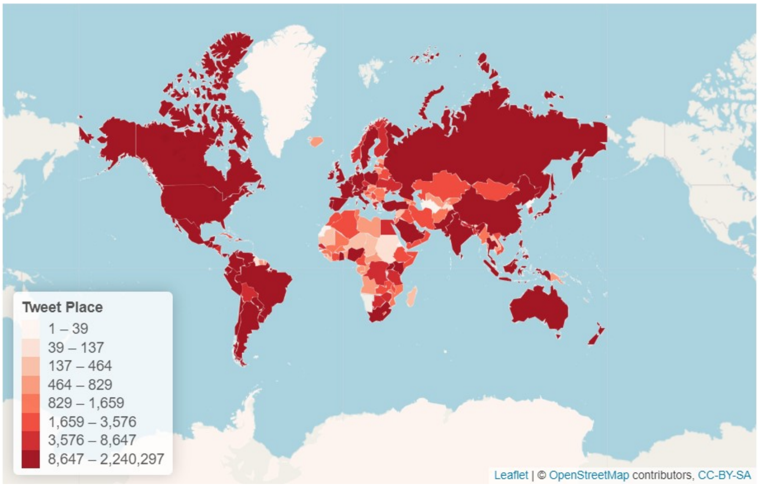


Figure 4.1: Location-tagged tweets from COVID-19 data *Adapted from:* [51]

The dataset was too big to deal with, so we split it by months and extracted only English tweets by applying a filter on the "lang" column provided. Then we saved only the tweets' IDs. Handling a dataset of this size was made feasible through the use of Vaex[2], a Python open-source DataFrame library that can easily handle datasets that would otherwise be too large to fit in RAM.

To get the raw tweet text from the tweet IDs, we selected a random sample of 100,000 tweets from 16 months starting from February 2020 till June 2021, and hydrated[3] them using Hydrator app[4], an Electron-based desktop application for hydrating Twitter ID datasets. The dataset obtained from Hydrator contained 35 features, but we worked on only two: "full_text", which is the full, raw text of the tweet; and "Datetime", which is the date and time when the tweet was posted.

TweetEval - Sentiment Analysis

Sentiment analysis is a complex and challenging problem. Yet in the end it can be thought of as a classification problem which needs a supervised learning classifier. Thus a labeled dataset is needed to learn patterns and be able to make predictions.

We used the benchmark SemEval 2017 - Sentiment Analysis in Twitter dataset from TweetEval [52], which consists of tweets annotated with either one of the three labels: positive, neutral, or negative. This dataset is commonly used to evaluate the performance of language models on Twitter data.

[2] https://towardsdatascience.com/vaex-out-of-core-dataframes-for-python-and-fast-visualization-12c102db044a

[3] https://stackoverflow.com/questions/34191022/what-does-hydrate-mean-on-twitter

[4] Documenting the Now. (2020). Hydrator [Computer Software]. https://github.com/docnow/hydrator

The dataset was initially used in SemEval-2017 Task 4, subtask A. Results of SemEval-2017 showed that the highest macro-average recall is 68.1%. Top teams used deep learning; one used an ensemble of LSTMs and CNNs with multiple convolution operations, while the other used deep LSTM networks with an attention mechanism [53]. The dataset was then released to the public to be used freely beyond SemEval. Five years later, *Loureiro et al.* [54] achieved a macro-average recall of 73.2% using BERT-based language models that are continuously trained on social media in order to investigate language model degradation and cultural shifts affecting language usage on social media.

The train, test, and validation sets were provided in separate files, but the train set was highly imbalanced. we concatenated train and test sets together, and under-sampled the data to offset the class imbalance. Then we used sklearn python library's train_test_split method to split the the data into train (80%) and test (20%).

4.1.2 Data Preparation

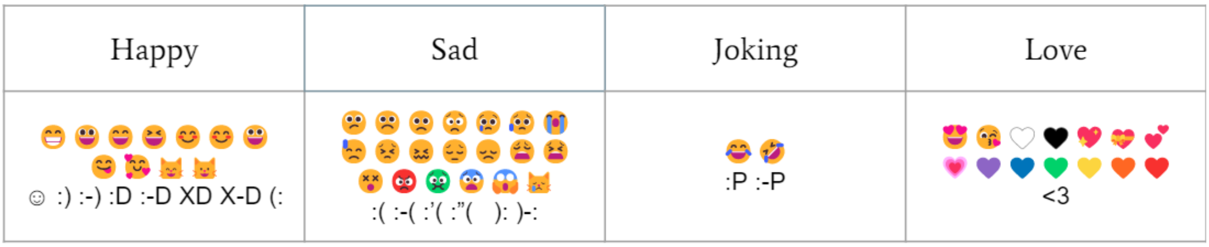
Here, we focused on cleaning and pre-processing both aforementioned datasets to remove the noise. This would decrease input size to reduce training time and boost the learning algorithm's accuracy. We performed the cleaning and pre-processing in two phases: phase 1 (P1) and phase 2 (P2).

The obtained dataset sample still contained non-English tweets. Thus, phase 1 (P1) of the cleaning process began with filtering out duplicates and non-English - as we are focusing on English tweets now - using *langdetect*[5] python library that detects the language of text.

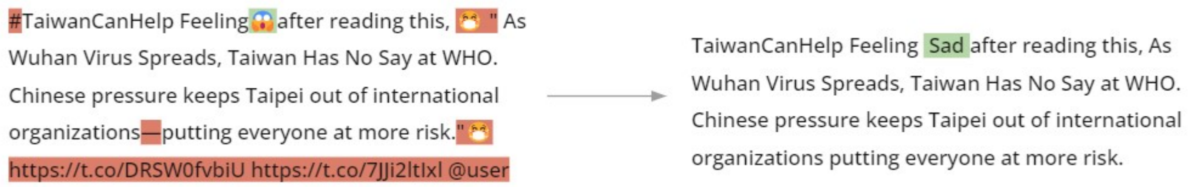
Non-textual contents and contents that are irrelevant for the analysis were identified and eliminated; this includes quotation marks ("") hashtag symbols (#), Twitter user mentions (@xxx), URLs, special Unicode and HTML characters like (&amp;); Twitter jargon like "RT" (representing re-tweet) and "QT' (representing quotes); and unnecessary white spaces and tabs.
Some researchers suggested the normalization of user mentions; replacing all user mentions with a word like "USER_MENTION" [55] or "@user" [56].
However, when we tested the current approach against theirs, we observed that removing user mentions achieved slightly better accuracy. Thus we chose to eliminate them completely, seeking to reduce input size as well.

In phase 1 (P1), we did not remove punctuations and kept the original case of words as it might affect the classification results for the sentiment analysis task. Similarly, we did not remove all emojis and emoticons since they can contribute to the score of tweet sentiments, as they convey diverse emotions. However, we replaced some emojis and emoticons with words that express the same mood, inspired by the approach of *Pota, M. et al.* [56], then removed every other emoji. The complete list of the emojis and emoticons considered in this thesis, which are the most frequently used, is shown in Table 4.1
An example of a tweet before and after the cleaning process for sentiment analysis is presented in Figure 4.2.

Table 4.1: Transformation of emojis and emoticons



Happy	Sad	Joking	Love
😁 😀 😄 😆 😊 😊 😃 😋 🥰 😺 😸 ☺ :) :-) :D :-D XD X-D (:	🙁 ☹️ 😟 😧 😢 😥 😭 😓 😣 😖 😔 😞 😩 😫 😵 😡 🤢 😨 😱 😿 :(:-(:'(:"( ):)-:	😂 🤣 :P :-P	😍 😘 🤍 🖤 💖 💝 💕 💗 💜 💙 💚 💛 🧡 ❤️ <3



#TaiwanCanHelp Feeling 😱 after reading this, 😷 " As Wuhan Virus Spreads, Taiwan Has No Say at WHO. Chinese pressure keeps Taipei out of international organizations—putting everyone at more risk." 😷 https://t.co/DRSW0fvbiU https://t.co/7JJi2ltlxl @user

TaiwanCanHelp Feeling Sad after reading this, As Wuhan Virus Spreads, Taiwan Has No Say at WHO. Chinese pressure keeps Taipei out of international organizations putting everyone at more risk.

Figure 4.2: An example of a tweet before the cleaning process (left) and after the cleaning process (right). The parts highlighted in red are removed, while the ones highlighted in green are replaced.

We built phase 2 (P2) upon P1, where we continued with converting all uppercase characters to lowercase. Punctuations were stripped, and alphanumeric words, stopwords, emojis, and emoticons were removed. This was achieved through the use of regular expression. Furthermore, we tokenized the tweets to N-grams: uni-grams, bi-grams, and tri-grams; and used WordNet lemmatizer from NLTK to lemmatize the tokens according to their parts-of-speech (POS) tags.

4.1.3 Sentiment Classification

Baseline Models

Before fine-tuning BERT for the classification task, we compared the performance of some machine learning models on the task of multi-class sentiment classification. The choice of the models was based on their exemplary performance mentioned in the previous work conducted in sentiment analysis. The following models were tested:

- One-vs-rest Logistic Regression (OVR) + TF-IDF as features (n-grams in the range of 1 to 2).
- Multinomial Logistic Regression + TF-IDF (n-grams in the range of 1 to 2).
- Random Forest (RF) + TF-IDF (n-grams in the range of 1 to 2).

[5] https://pypi.org/project/langdetect/

- Multinomial Naïve Bayes (MNB) + CountVectorizer.
- A neural network with 1 LSTM layer with 0.2 dropout rate + GloVe word embeddings.
- A neural network with 2 LSTM layer with 0.2 dropout rate + GloVe word embeddings.
- A neural network with Bi-LSTM layer with 0.5 dropout rate + GloVe word embeddings.
- A neural network with Bi-LSTM layer with 0.5 dropout rate + 1 Attention layer + GloVe word embeddings.

All tweets of the TweetEval's Sentiment Analysis dataset underwent both data preparation phases (P1 and P2), and then the models were trained and tested on the them.

Model evaluation metrics include accuracy, precision, recall, and F-score. Accuracy is a fraction of the number of correct predictions over the total number of predictions.

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{4.1}$$

Recall represents a fraction of how many instances were classified correctly as positive to the total number of true positives and false negatives. It shows what proportion of actual positives was identified correctly.

$$Recall = \frac{TP}{TP + FN} \tag{4.2}$$

Precision shows the fraction of the number of instances the model classified as positive to the total number of instances classified as positive. It shows the correctness of the classifier with respect to each class.

$$Precision = \frac{TP}{TP + FP} \tag{4.3}$$

Finally, F-score combines rrecision and recall. *Rijsbergen* [57] defined it as the harmonic mean of precision and recall.

$$F = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4.4}$$

We evaluated our models by means of two metrics: classification accuracy and recall.

BERT Fine-tuning

We used the HuggingFace Transformers library[6], which provides a PyTorch interface to fine-tune pre-trained language models. We specifically used $\text{BERT}_{\text{Base-cased}}$ pre-trained model, which is case sensitive.

Before passing the input sequences to BERT encoder, we needed to specify the maximum length of our sentences for padding/truncating to. So, after P1 of the data preparation step, we performed one tokenization pass of the dataset to store the lengths of each tokenized tweet, then we plotted the distributions only to find that most of the tweets contained less than 70 tokens, as shown in Figure 4.3. To be on the safe side, we set the maximum length to 85.

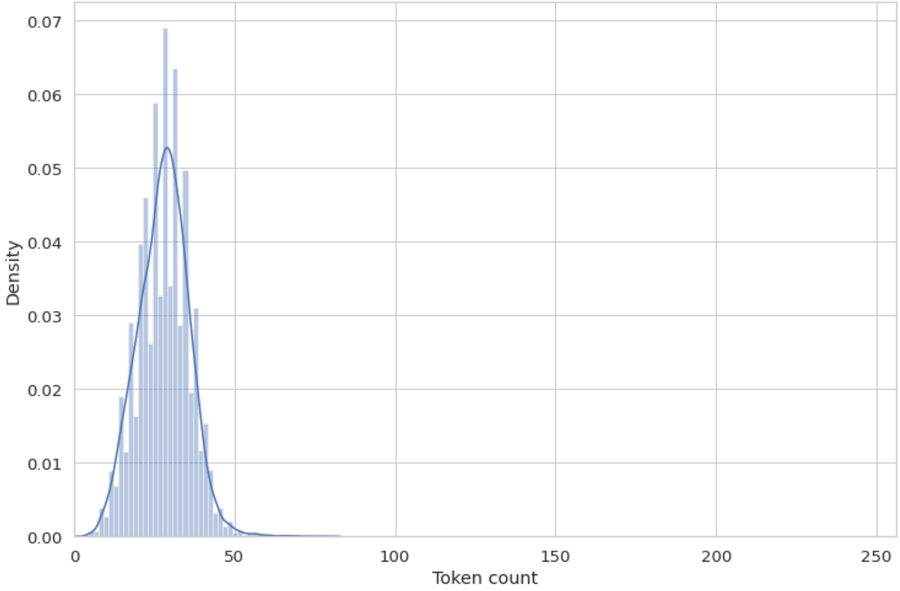


Figure 4.3: Distribution of tokenized tweets lengths in TweetEval's SemEval 2017 - Sentiment Analysis in Twitter dataset

For fine-tuning BERT, we used the model described in section 3.1.3, where we add a single-hidden-layer feed-forward neural network on top of BERT, with a dropout rate of 0.3. We experimented with different hidden layer sizes, specifically: 50, 128 and 512, but all have yielded the same results. The experiments conducted were manual; we ran the whole model three times each with a different hidden layer size, and compared the final results.

The optimal hyper-parameter values are task-specific, but, the authors recommend some hyper-parameters for fine-tuning that they found they work well across almost all tasks. Table 4.2 shows the recommended values.

[6] https://huggingface.co/docs/transformers/index

Table 4.2: Recommended hyper-parameter values. *Adapted from: [26]*

Hyper-parameter	Values
Batch size	16 or 32
Learning rate (Adam)	5e-5, 3e-5 or 2e-5
Number of epochs	2, 3, or 4

We manually tested different combinations of these values, including a learning rate of 1e-5. Table 4.3 shows the hyper-parameter combination that achieved the highest accuracy, according to our experiments.

Table 4.3: Optimal hyper-parameters for the sentiment analysis task.

Hyper-parameter	Value
Batch size	16
Learning rate (Adam)	1e-5
Number of epochs	3

4.1.4 Sentiment Trends

For data pre-processing, P1 is applied to prepare the COVID-19 tweets dataset to generate predictions. We then proceeded to apply the fine-tuned model on the tweets.

We observed that among the tweets in our dataset, the proportion of neutral tweets is the highest, forming around 46% of the whole dataset. This is followed by the proportion of negative tweets, which comprised about 39% of total tweets, as illustrated in Figure 4.4.

To examine the trends of the sentiment over time and derive a sentiment arc, we experimented with different methods:

1. Trailing moving average.
2. Taking the mean of daily sentiments.
3. Take the mean of daily sentiments then apply Savitzki-Golay filter.

The purpose of the first two experiments is to investigate whether taking the daily average would affect the resulting pattern drastically. The third method aims to fit a smooth trend line since the daily averages of tweet sentiment are highly noisy.

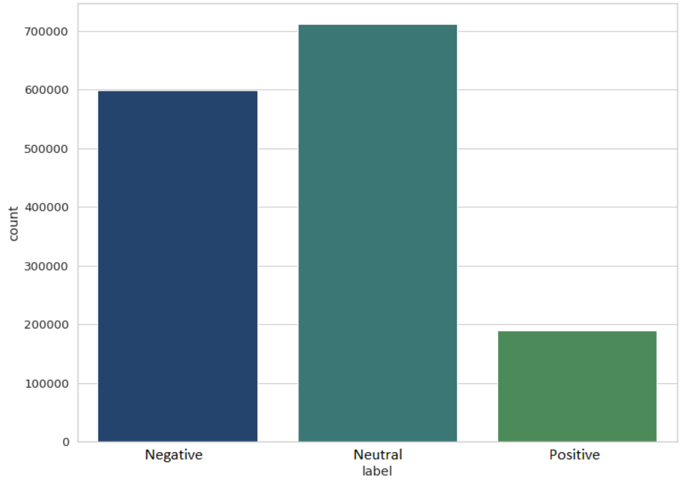


Figure 4.4: Proportion of each sentiment class in the COVID-19 dataset.

4.1.5 Topic Modeling

We explored topics discussed in tweets using GSDMM. We experimented with setting the upper bound of the GSDMM with a different number of topics. However, we finally chose a model with 9 topics among all models because it showed diverse and less redundant topics when manually examined. The alpha and beta parameters are set to 0.1 as used in the original paper, and the number of iterations is set to 30 as GSDMM does not require a high number of iterations to converge [49].

We applied the GSDMM algorithm on tweets posted during three stages of the pandemic: the start of the outbreak (February - April 2020), the second wave (July - September 2020), and the start of the vaccination campaign (December 2020 - February 2021).

4.2 Results

4.2.1 Model performance

The performance results of all the tested models on the TweetEval dataset are shown in Table 4.4 The results of the baseline models are not considerably different when compared to one another. However, the BERT$_{\text{Base}}$ approach exhibited a significant improvement.

The confusion matrix for our model is reported in Table 4.5.

Table 4.4: Experimental results on TweetEval dataset.

Model	Test Accuracy	Macro-recall
OVR + TF-IDF	0.65	0.61
Multinomial regression + TF-IDF	0.66	-
Random Forest + TF-IDF	0.64	0.58
Multinomial Naïve Bayes + CountVectorizer	0.62	0.60
1 LSTM + GloVe (15 epochs)	0.66	-
2 LSTM + GloVe (25 epochs)	0.66	-
Bi-LSTM + GloVe (15 epochs)	0.64	-
Bi-LSTM + Attention+ GloVe (15 epochs)	0.64	-
$\text{BERT}_{\text{BASE}}$ + FFNN (3 epochs)	0.74	0.74

Table 4.5: Confusion matrix for the sentiment analysis task performed using $\text{BERT}_{\text{Base}}$.

	Predicted Negative	Predicted Neutral	Predicted Positive
Actual Negative	850	194	53
Actual Neutral	221	855	200
Actual Positive	46	216	926
Recall	0.77	0.67	0.78
Precision	0.76	0.68	0.79

To verify the results, we re-ran the BERT model multiple times, on different train test subsets by specifying a different random_state on splitting the data. The lowest testing accuracy and macro-average recall achieved were 0.72 and 0.73 respectively.

4.2.2 Sentiment Dynamics and Time Series Analysis

Regarding the first method taken to investigate the sentiment trends, a moving average with a window size of 450 was chosen due to multiple experiments that aimed to distinguish a general trend from the data. Figure 4.9 shows the plot, which is still very noisy.

Figure 4.10 shows the time series of average sentiment for Twitter, binned by day. The plot is less noisy than the previous one, and a general trend can be seen. A smoothed-out trend is plotted in Figure 4.11 to help better see patterns in the time series that otherwise could have been obscured by the seasonal patterns. The term “seasonal” pattern does not indicate a meteorological season here. Instead, it refers to a repeating pattern with a fixed length in the data.

When examining the trend, we observe that it is overall negative. However, the scores start badly with a low sentiment score that sinks further around the end of February, when the Centers for Disease Control and Prevention (CDC) announced that COVID-19 is heading toward pandemic status, as it has met 2 of the three required factors: illness resulting in death and person-to-person transmission. Worldwide spread is the third criteria not yet met at that time[7]. Feelings of fear and worry might explain the initial trend. Fear is a common occurrence for people exposed to a worrying infectious disease [58], and might have been worsened by the often inadequate information and the uncertainty surrounding the pandemic at that time; when people did not know how they would be impacted, how long this would last or how bad things might get.

Later on March 11, 2020, the World Health Organization (WHO) declared the COVID-19 outbreak a global pandemic. This has led governments worldwide to respond unprecedentedly to contain the virus. However, the lockdown regulations appeared to be associated with lower sentiment scores. For example, lack of mobility, reduced access to entertainment activities, and concerns about school or jobs decreased people's happiness as measured on social media [59]. In the first few weeks of lockdown, an increase in search intensity for boredom, sadness, loneliness, and worry was reported in Europe, and the United States [60].

Many countries have experienced a two-wave pattern in the reported cases: the first wave during the spring of 2020 and the second in the late summer of 2020. This might explain the decreased sentiment levels at that time. However, the anticipated negative correlation between the sentiment scores and death rates is inconsistent; using *Pearson's R*, the correlation coefficient between the average daily sentiment scores and the daily death tolls reported by WHO was estimated to be 0.65202660, which suggested a positive correlation. This is also illustrated in Figure 4.12, which compares the daily death tolls due to COVID-19 with the averaged daily sentiment scores.

The results might initially appear paradoxical: the average sentiments shown by people leaned towards the neutral sentiment when the number of cases and deaths attributed to COVID-19 continued to rise in 2021. After consulting the literature addressing human behavior and psychology, we identified the observed pattern of the initial rise in distress followed by a gradual return to baseline levels as a common response in research examining adaptation to other types of major life stressors [61]. The process of "normalization" could also explain the pattern: normalization occurs when new social and policy practices and ideas developed during a crisis become the "new normal" and are taken for granted or dealt with as natural in everyday life [62].

Figures 4.8a and 4.8b are word clouds depicting the most common words among COVID-19-related tweets posted between February 2020 to April 2020; and July 2020 to September 2020, respectively. Pandemic-related words like "covid", "coronavirus" or "pandemic" were filtered out from the text before plotting the word cloud.
Discussions at the start of the pandemic revolved around China and Wuhan, the source of the outbreak; Donald Trump, the US president at that time; and general words commonly used during the pandemic's spread like "outbreak", "new", "quarantine" and "lockdown".

[7] https://www.ajmc.com/view/cdc-warns-that-covid19-is-likely-headed-toward-pandemic-stage-could-affect-us-schools-businesses

Except for tweets containing news headlines, people might have been expressing their fear and doubts while still getting adjusted to the new living situations imposed by the pandemic. Other tweets might have expressed anger towards the US president's denial and mismanagement[8]. Around the time the second wave started, we notice an increase in the frequency of the phrase "new case" in Figure 4.8b, which might refer to the increased number of reported cases. Words like "vaccine" and "testing" could be seen, due to the discussions addressing the vaccine clinical trials.

Vaccinations

We looked more into the impact of other possible factors; if the news in 2020 focused on COVID-19's global spread, the focus in 2021 has been on ending the pandemic through vaccine distribution. In December 2020, the FDA gave emergency use authorization to two COVID-19 vaccines: the Pfizer-BioNTech and the Moderna. In 2021, even more vaccines were approved. Therefore, we investigated how the people's collective attitudes towards the vaccines changed from December 2020 till the end of April 2021.
Figure 4.8c is a word cloud depicting the most common words among COVID-19-related tweets posted between December 2020 till the end of March 2021. It further supports that the population's focus turned towards vaccinations, where words such as "vaccine" and 'vaccinations" appear to be discussed more frequently. It is important to mention that stop-words and pandemic-related words like "covid", "coronavirus" or "pandemic " were filtered out from the text before plotting the word cloud.

We used a dataset of global tweets about the COVID-19 vaccines that is available on Kaggle[9]. The tweets were collected using the following vaccines as keywords: Pfizer/BioNTech, Sinopharm, Sinovac, Moderna, Oxford/AstraZeneca, Covaxin, and Sputnik V. Additionally, we filtered tweets from the COVID-19 dataset that contained these same keywords. We applied the sentiment analysis model to both datasets and observed that the dataset taken from Kaggle had 67,052 tweets, 20.7% of which are negative and 20.95% are positive. The other dataset contained 21.4% negative and 10.2% positive tweets out of 15,223 tweets. However, the overall average sentiment of each dataset was higher than the sentiments towards COVID-19 in general, as shown in Figure 4.13. Therefore, we could infer that the sentiment arc for COVID-19 was influenced by the introduction of vaccines, or other factors that have not been investigated yet.

Contextual Meaning of Words

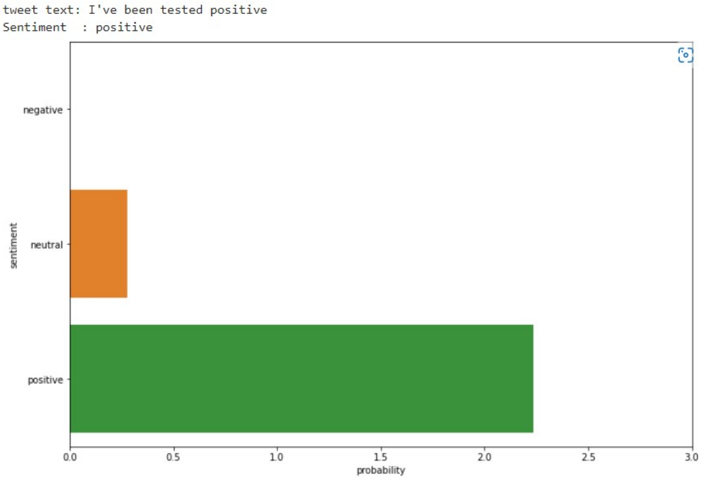
In linguistics, context carries tremendous importance in the disambiguation of meanings and understanding the actual meaning of words [63]. It is also evident that context could change a word's polarity in sentiment analysis.

[8] https://www.washingtonpost.com/graphics/2020/politics/trump-covid-pandemic-dark-winter/

[9] https://www.kaggle.com/datasets/gpreda/all-covid19-vaccines-tweets

The word "positive" usually means something pleasant or helpful when describing a fact, situation, or experience[10], and it implies a positive sentiment. However, in the health sector, a positive test result means that the substance or condition being tested for was found, and it should imply a negative sentiment.

The COVID-19 dataset is full of "positive" in the context of diseases, thus tweets like "I've been tested positive" should be classified as negative. However, since our model was pre-trained and fine-tuned on a general domain corpus, it would classify such example as positive. Figure 4.5 shows the classification result when we tested our model on "I've been tested positive" .



```
tweet text: I've been tested positive
Sentiment  : positive
```

Figure 4.5: The results of testing our sentiment analysis model on an input.

To see if that would influence the sentiment arc, we tested an approach where we replaced the word "positive" with "infected" in all tweets in the corpus that do not contain the words "impact" or "effect". Similarly, we replaced "negative" with "free". We have chosen "infected" and "free" because they would be classified as negative and positive, respectively. Figure 4.14 shows a plot of the sentiment arc resulting from this approach in comparison to the original trend line. The difference between the arcs is almost unnoticeable.

Happiness Scores

We carried out further analysis using the Hedonometer, previously mentioned in section 2.3. While it is a fact that the models measure different qualities of the text in different ways and on different scales, we were curious to see if the two arcs - one based on happiness scores and one based on sentiment scores - would exhibit roughly the same pattern.
Figure 4.15 shows both arcs on the same plot, each with different scoring scales: the Hedonometer scale is 1 (very negative) to 9 (very positive), with 5 being neutral; our

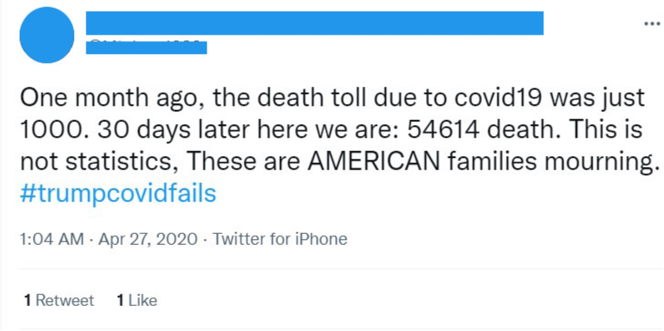
[10] https://www.collinsdictionary.com/dictionary/english/positive

scale is -1 (very negative), +1 (very positive) and 0 denotes neutral. Both arcs convey the overall daily average scores. Figure 4.16 shows an isolated arc of the Hedonometer to see the fluctuations in more detail.
Looking at the finer details of the arc generated by the Hedonometer, we observe a general similarity with the sentiments arc: the arc starts with the lowest happiness scores compared to the rest of the arc, then takes an overall upwards trajectory. On the other hand, the broader picture shows that the changes in happiness scores are very subtle; the trend seems to plateau.

4.2.3 Topic Modeling

We obtained 9 clusters from the GSDMM model for each of the three stages and used our judgment to assign labels to each cluster (topic). This was done by checking the list of words with a high probability of belonging to each topic and manually inspecting samples of tweets per topic to get a better sense of the content. Finally, we assigned a label to each topic. The set of labeled topics is presented in Table 4.6 for the first stage (February - April 2020), Table 4.7 for the second stage (July - September 2020), and Table 4.8 for the final stage (December 2020 - February 2021).

We observed that at the start of the pandemic, the most discussed topics included people commenting on the former US president's initial denial of the outbreak, accusing him of being a liar after some announcements he made. Figure 4.6 shows an example of such tweets. Other topics were the expression of support towards front-line workers; discussions on preventive measures or impact on the economy; expression of sadness towards canceled plans; and updates on new COVID-19 cases and deaths.



[redacted]

One month ago, the death toll due to covid19 was just 1000. 30 days later here we are: 54614 death. This is not statistics, These are AMERICAN families mourning. #trumpcovidfails

1:04 AM · Apr 27, 2020 · Twitter for iPhone

1 Retweet 1 Like

Figure 4.6: A tweet sample.

At the start of the second-wave, government announcements were still among the most discussed. Other topics that evolved were revolving around the US presidential election campaigns as the US presidential elections were approaching. Discussions about vaccination trials came up as well. Discussions about the impact of the pandemic on the economy and on mental health are still consistent, as well as the discussions about preventative measures and restrictions. However, the latter category was discussed less frequently than before.

As the year 2021 began and 2020 was ending, “year” appeared in many discussions referencing to the start of a new year. Gratitude for having the family around, and for surviving a 2020 with all the challenges it brought, were frequently mentioned as well. This was also around the time that vaccination campaigns began, hence vaccinations were among the most frequently debated topics, as shown in topics 4 and 9 4.8. The tweets in the topic 5 cluster expressed concern about a variety of issues, including vaccines and new variants of COVID-19.

Table 4.6: Topics discussed during February - April 2020 and their labels. Ordered by their frequency.

No.	Label	Top Words
1	Government actions & response	people, trump, virus, world, government, country, pandemic, spread, die, lie
2	Expressing support for healthcare/front-line workers	help, work, support, health, thank, fight, people, crisis, pandemic, business
3	Stay home/Preventive measures	people, spread, case, home, test, quarantine, virus, health, lockdown, stay
4	Hopes to find a vaccine	virus, people, test, patient, vaccine, flu, spread, case, die, symptom
5	Gratitude for time at home with loved ones	people, stay, home, thank, love, family, die, work, life, help
6	Preventive measures	people, hand, virus, mask, spread, wash, stay, home, work, food
7	New cases & deaths reports	case, death, confirm, report, number, update, country, people, infection, rise
8	Impact on economy	market, impact, economy, fear, hit, stock, spread, business, outbreak, crisis
9	Cancelled plans or events	cancel, postpone, event, year, concern, game, week, news, travel, world

Table 4.7: Topics discussed during July - September 2020 and their labels. Ordered by their frequency.

No.	Label	Top Words
1	Government announcements	people, die, trump, life, death, virus, family, year, tell, world
2	US presidential election	trump, people, lie, vote, death, die, election, virus, president, test
3	Vaccine trials	vaccine, test, virus, people, patient, study, death, treatment, trial, health
4	Work from home	test, school, student, case, testing, people, health, home, spread, week
5	Jobs crises & mental heath	pandemic, impact, help, work, support, health, business, read, thank, community
6	New cases & deaths reports	case, death, report, number, confirm, test, update, record, today, state
7	Impact on economy	pandemic, government, people, help, business, economy, job, year, pay, crisis
8	Lockdown & restrictions	test, case, restriction, lockdown, people, news, week, game, player, season
9	Preventive measures	mask, spread, help, wear, people, face, thank, work, hand, home

Table 4.8: Topics discussed during December 2020 - February 2021 and their labels. Ordered by their frequency.

No.	Label	Top Words
1	Government actions towards vaccines	people, vaccine, death, government, die, year, country, state, pandemic, trump
2	Gratitude for time at home with loved ones	year, people, vaccine, love, thank, today, life, home, family, feel
3	Lockdown & restrictions	case, school, people, mask, restriction, spread, lockdown, travel, test, state
4	Vaccines	vaccine, dose, vaccination, receive, people, country, vaccinate, shot, week, health
5	Sharing concerns about different topics	vaccine, virus, people, patient, study, risk, test, variant, death, infection
6	Expressing gratitude & support	pandemic, health, help, support, work, vaccine, people, year, community, thank
7	New cases & deaths reports	case, death, report, number, update, today, confirm, record, infection, test
8	Impact on economy	pandemic, business, impact, year, help, vaccine, work, read, market, report
9	Vaccines	vaccine, vaccination, appointment, test, site, today, people, receive, testing, health

4.3 Application of the Model on Other Domains

To prove that the proposed model is generic, we applied it to tweets from another domain.
While working on the thesis, on April 14, 2022, the business magnate Elon Musk announced that he is offering to purchase Twitter after previously acquiring 9.1 percent of the company's stock. While some people celebrated the deal in the name of free speech, others were worried about the platform's future. This posed an ideal opportunity to track the collective sentiment of the public.

Using Twitter API[11] and *Tweepy* Python library, We collected 942,612 tweets over almost 16 days. The collection process started on April 15, 2022, using the query "(elon AND Twitter) OR Elon Musk" On April 26, we added the keywords "TwitterTakeover" and "ElonMuskBuysTwitter" as they started trending on that day after Twitter's board publicly accepted the offer on April 25.

To prepare the data for the sentiment analysis model, we apply phase 1 (P1) of the pre-processing and cleaning step. Figure 4.7 shows that negative tweets are dominant in this dataset. Finally, Figure 4.17 shows the sentiment arc of tweets concerning Elon Musk buying twitter after taking a average of tweets' sentiments in every 6 hours and applying a Savgol smoothing filter with window size = 15.

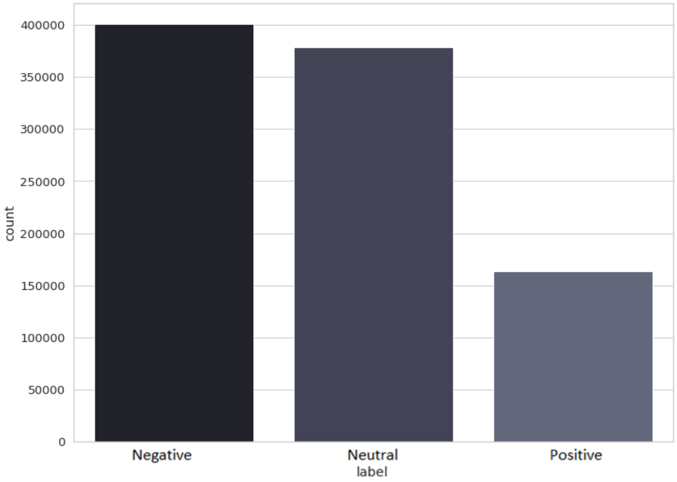


Figure 4.7: Proportion of each sentiment class in the dataset

People started with a negative outlook, but perhaps gradually got in terms with it. If we attempt to explain the arc, we observe a break in the upwards trajectory of the trend between the 23rd and 25th of April. After a manual survey of the tweets posted that day, we could extrapolate that this break is the public's response to 2 tweets Elon has posted mocking another entrepreneur publicly. The positive peak on the 26th of April might indicate that people were changing their viewpoint after Twitter's board accepted the

[11] https://developer.twitter.com/en/docs/twitter-api

buyout offer. Alternatively, it might be attributed to a surge of tweets from supporters of the deal.

4.4 Limitations

Our approach performs sentiment annotation at the document-level, which means that all entities in a tweet are considered to have an equal sentiment polarity. For instance, if a tweet mentions two vaccines, where one is the sentiment holder while the other is not, our approach assigns the same polarity to both entities.

We know that social media users, in general, are a non-representative sub-population of all people. In addition to the small sample of tweets we worked on, these might not represent the world's general population fairly.

Furthermore, we only quantify how people appear online; our method does not determine an individual's internal emotional states. To truly understand a social system's potential dynamical evolution, one should account for publicly hidden but accessible internal ranges and states of emotions, beliefs, and so on [30].

Lastly, text analysis techniques have difficulty detecting irony or sarcasm, which can be found frequently in social media posts. In addition, social media posts tend to contain misspellings, slang terms, and shortened forms of words, all of which can affect the accuracy of a sentiment classification model. Still, the tracking system is useful for generating a visual sketch of the population's mood.

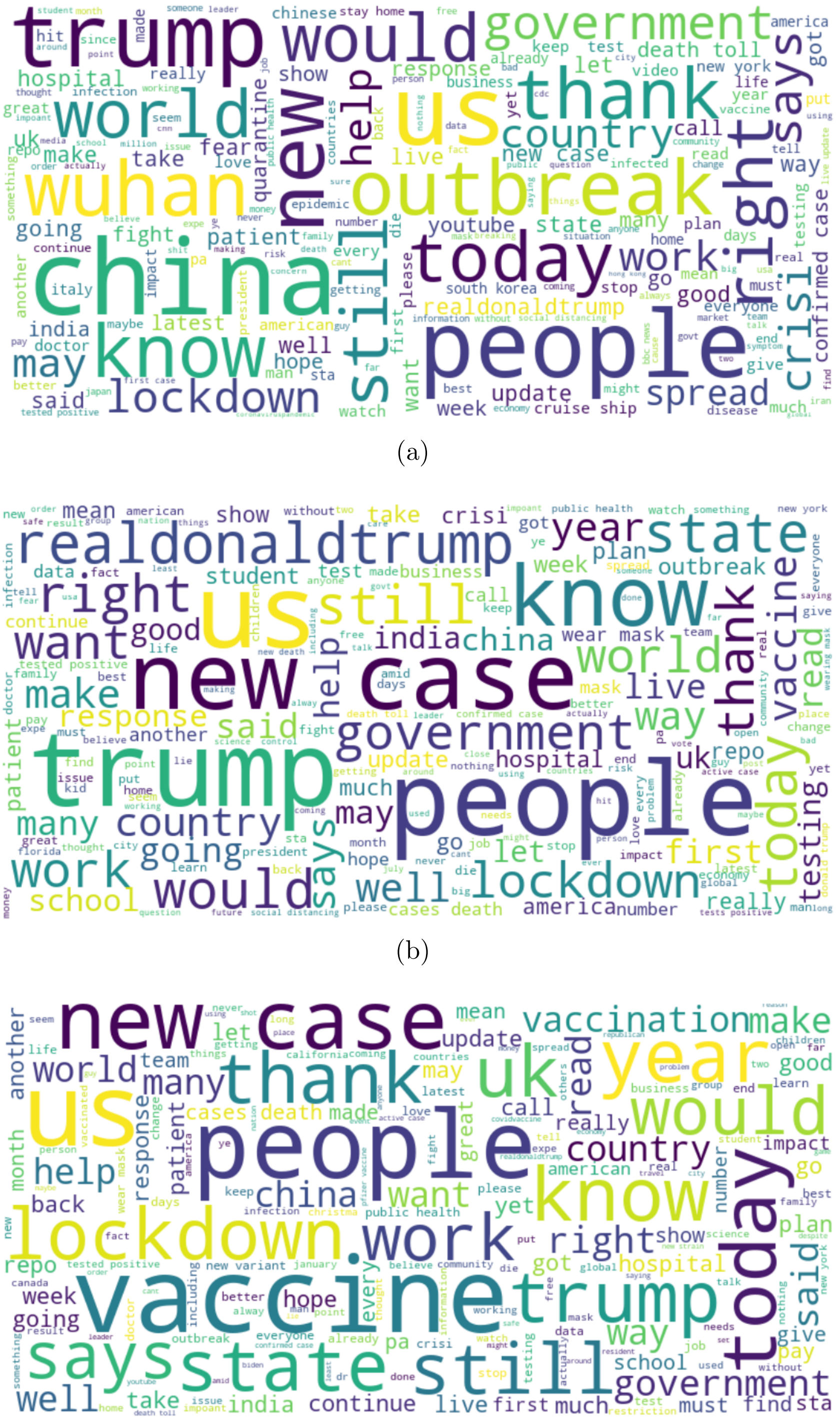


Figure 4.8: Word clouds for top words among tweets posted during three stages of the pandemic: the start of the pandemic (a), the start of the second wave (b), and start of the vaccination campaign (c).

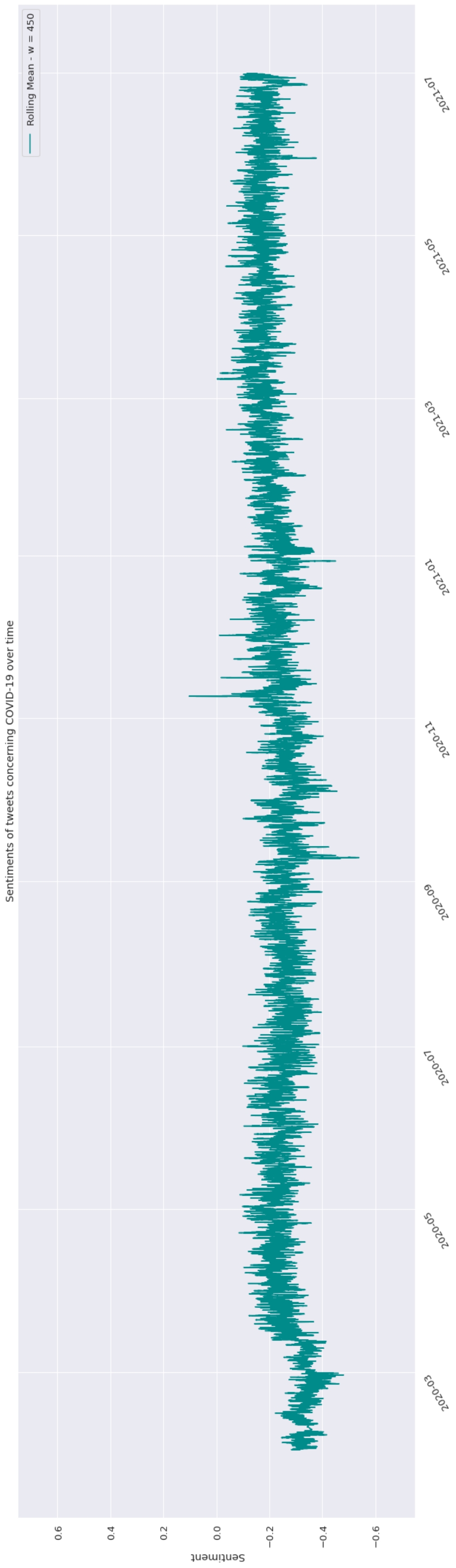


Figure 4.9: Sentiments of tweets concerning COVID-19 over time using moving average smoothing.

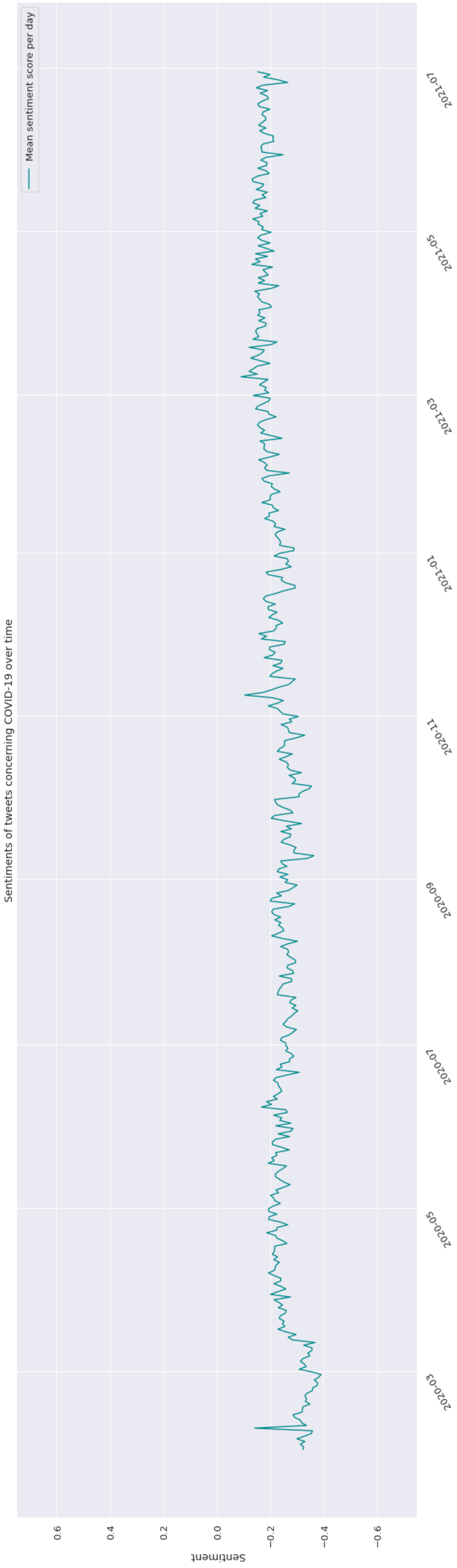


Figure 4.10: Average daily sentiments of tweets concerning COVID-19 over 17 months.

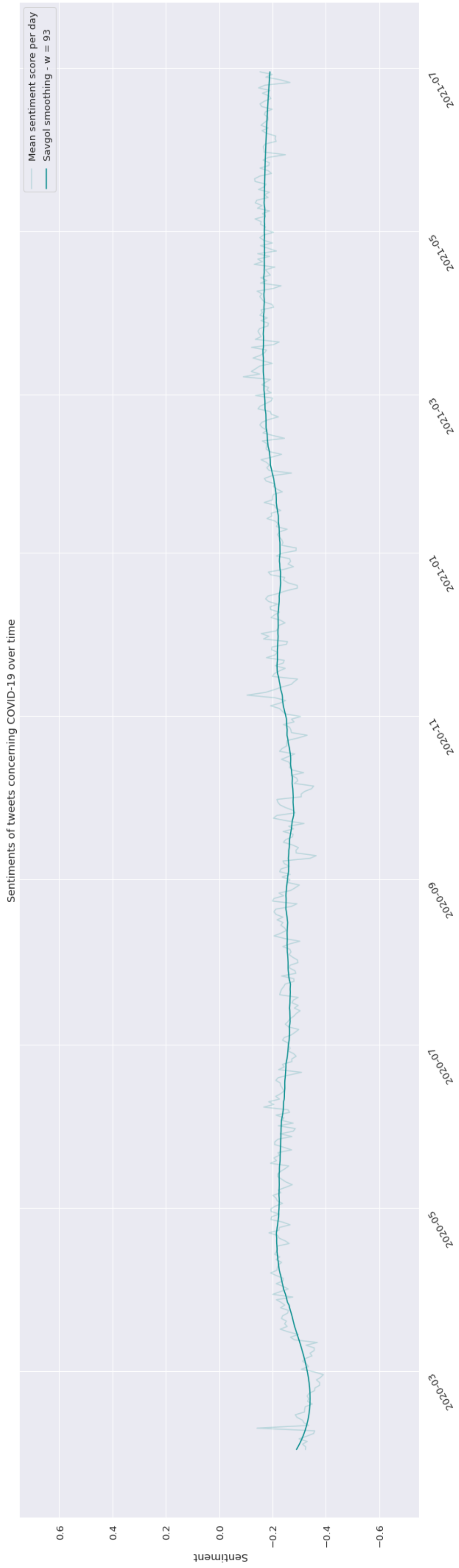


Figure 4.11: Average daily sentiments of tweets concerning COVID-19 over 17 months using Savgol filter for smoothing.

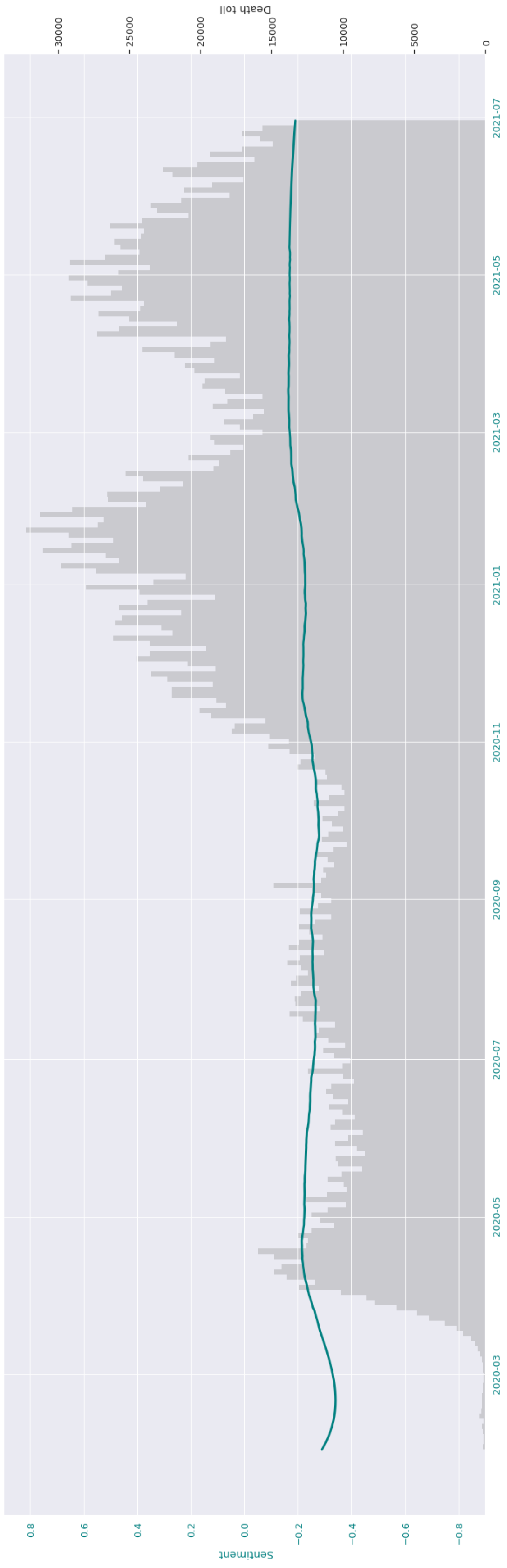
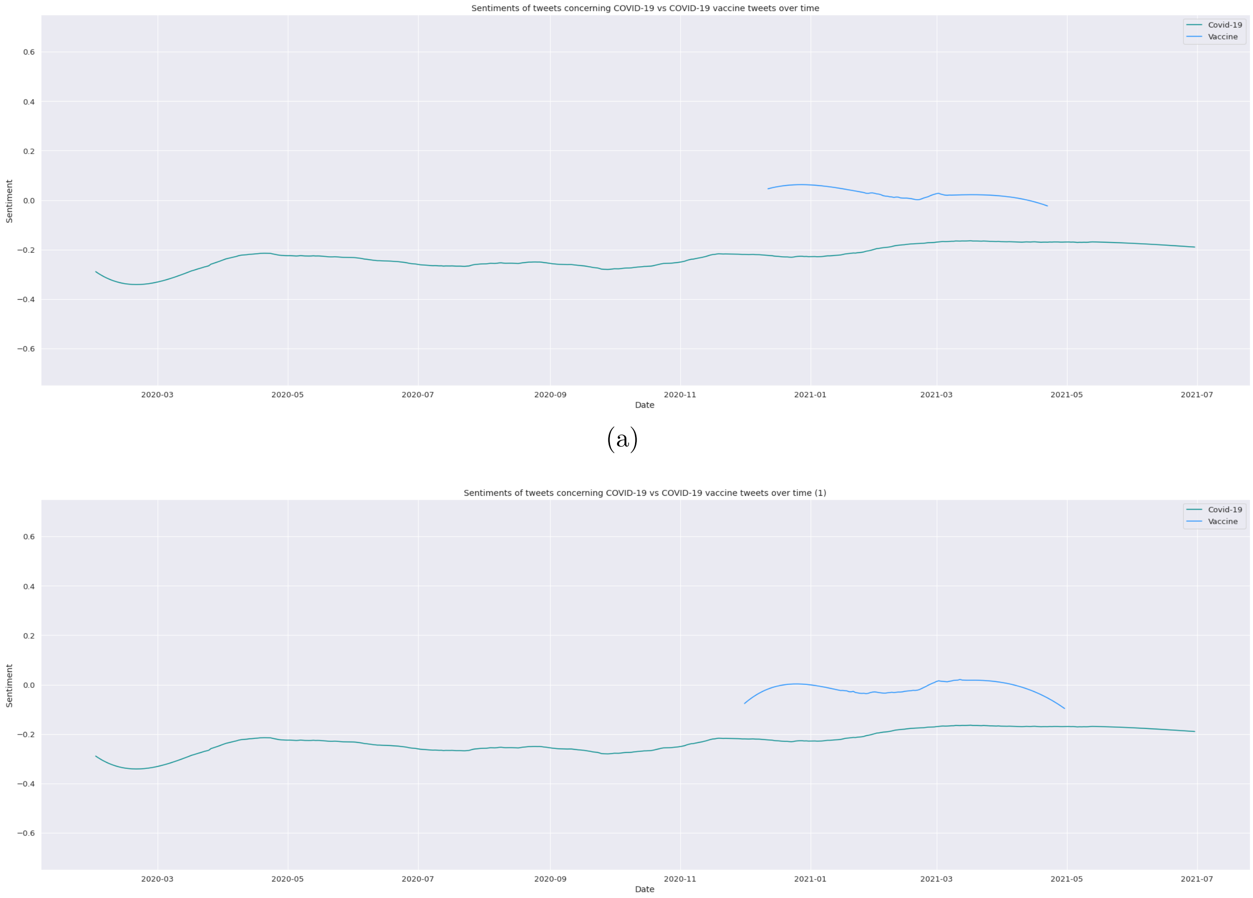


Figure 4.12: Smoothed average daily sentiments of tweets concerning COVID-19 over 17 months in comparison to the death tolls.



(a)

(b)

Figure 4.13: Average daily sentiments of tweets concerning COVID-19 (green), in comparison to the average daily sentiments towards different vaccines (blue). Both arcs are smoothed using Savgol filter. (a) illustrates the sentiments of tweets taken from the COVID-19 dataset. (b) illustrates the sentiments of tweets taken from Kaggle.

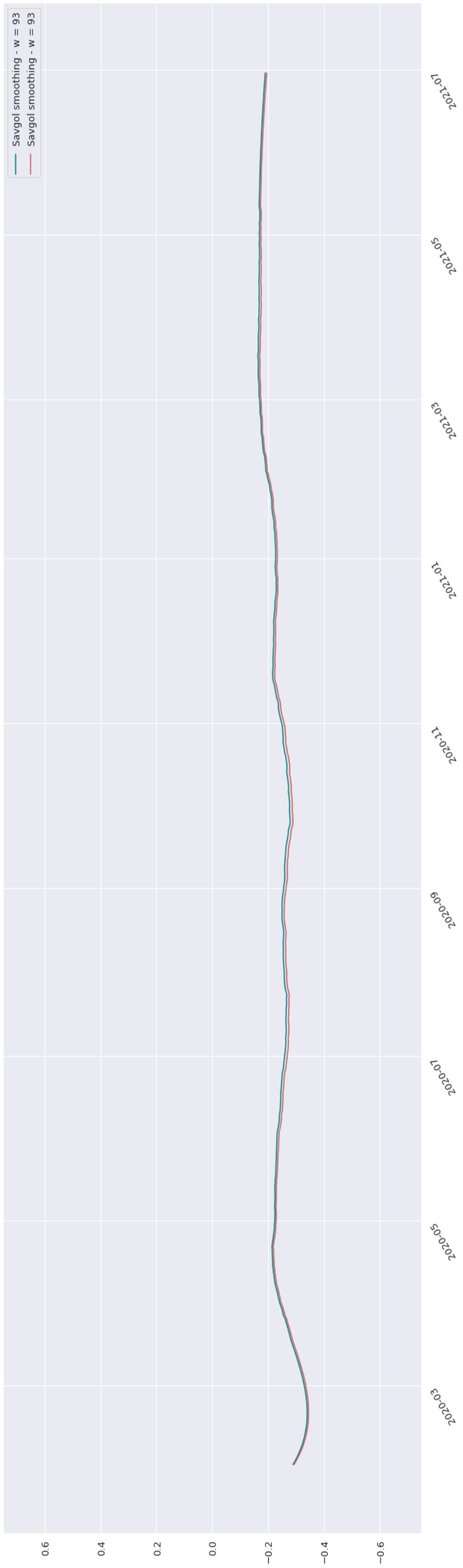


Figure 4.14: Comparison of original sentiment arc COVID-19 tweets (green) with the sentiment arc generated from replacing “positive” with “infected” and “negative” with “free” (green).

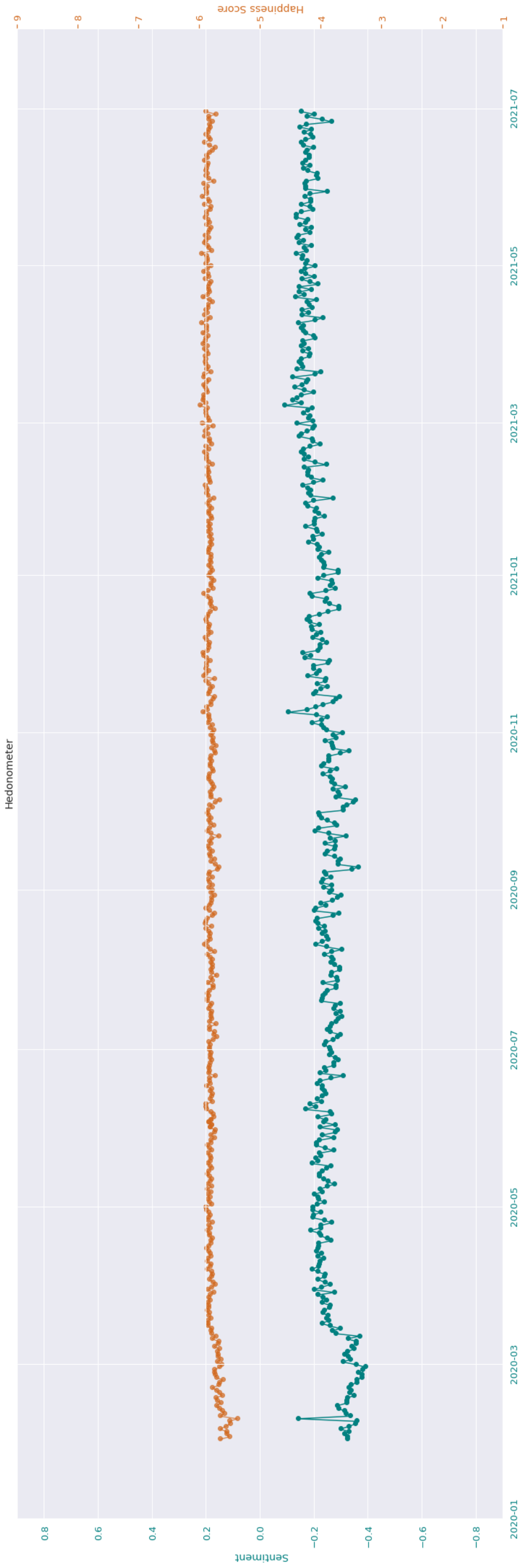


Figure 4.15: Average daily happiness scores of tweets concerning COVID-19 (orange) vs average daily sentiment scores of tweets concerning COVID-19 (teal).

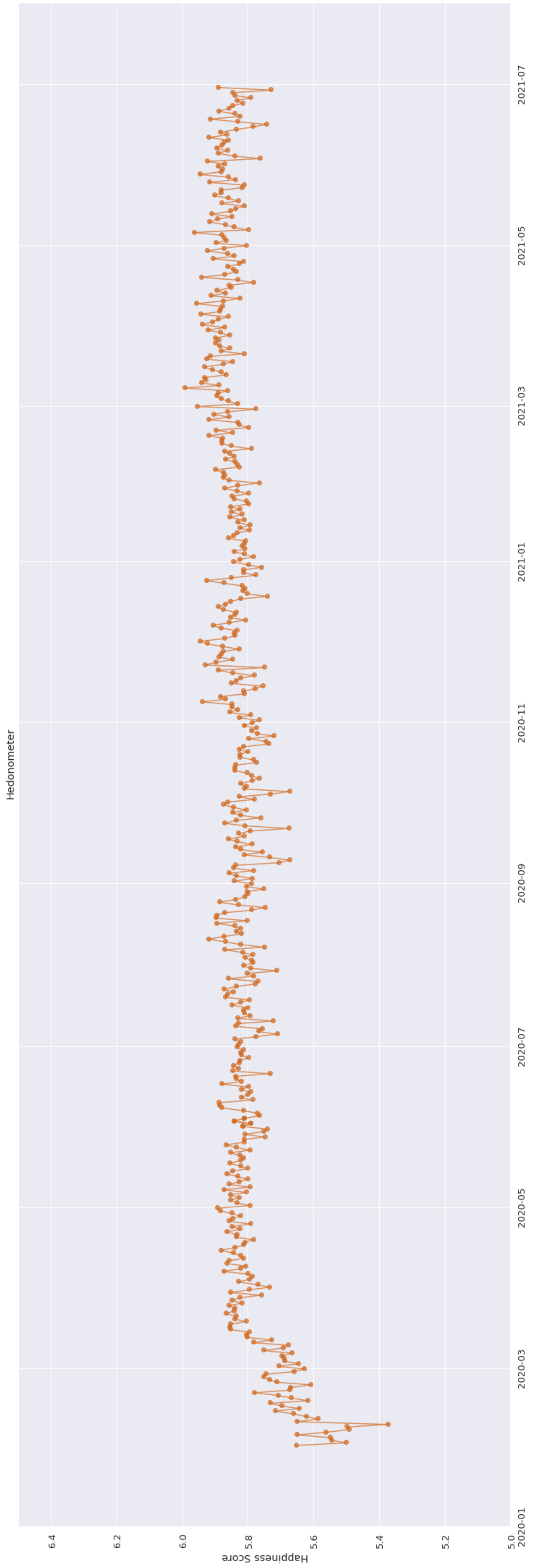


Figure 4.16: Average daily happiness scores of tweets concerning COVID-19 using the Hedonometer

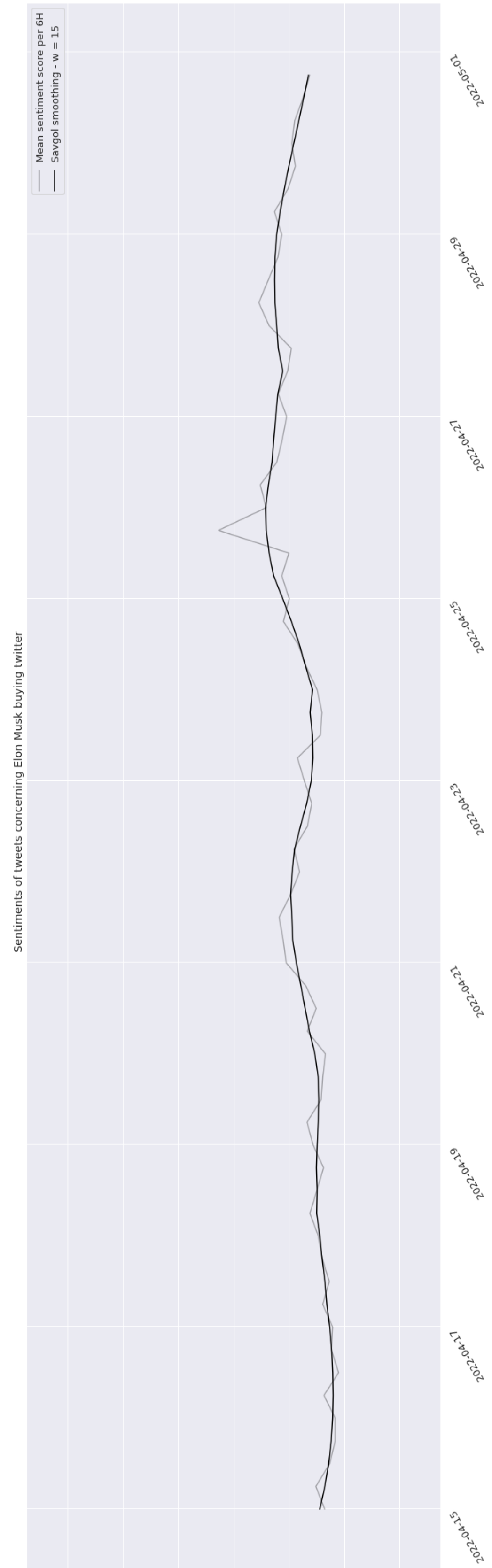


Figure 4.17: Sentiment arc of tweets concerning Elon Musk buying twitter.

Chapter 5

Conclusion & Future Work

5.1 Conclusion

This thesis aimed to study and analyze sentiments revealed by people through their social media posts about a major event and track how the sentiments change over time through the construction of a sentiment arc. We defined an architecture composed of $\text{BERT}_{\text{Base}}$ followed by a final classification layer (single-layer FFN with softmax activation), and we fine-tuned the model for document-level sentiment analysis. We measured the model's performance by applying it to a benchmark dataset for sentiment analysis and obtained a test accuracy of 74%, from which it was possible to deduce that BERT's language modeling power significantly contributes to achieving good results.

We used the model to classify time-stamped tweets concerning the COVID-19 pandemic into positive, negative, or neutral. We annotated them with intensity scores, then plotted the temporal trends using a sliding window technique with Savitzky-Golay filter. We observed that the sentiment arc started at a low point at the beginning of the outbreak but gradually leaned towards the neutral. Furthermore, we investigated possible events that might have influenced resulting sentiment arc. The arc was compared to one we plotted for the COVID-19 vaccines and to the death rates released by WHO. Key topics and events impacting the trend were also identified, evaluated, and mapped to the temporal trends. We also consulted the psychological research and literature, and learned that the resulting pattern is a typical response to major life stressors. Further experiments were conducted to compare the sentiment arc to an arc generated by an instrument that measures the happiness scores in text: the Hedonometer.
We also investigated the effect of replacing the words "positive" and "negative" in the COVID-19 tweets corpus to "infected" and "free", respectively. This was done to account for the fact that our model was trained on a general domain data, and thus would assign "tested positive" a positive sentiment. The difference between the resulting arcs was very subtle.

All experiments were carried out to determine how accurate the arc is in comparison to actual events, and they revealed that the arc did present a degree of actuality. To prove that our sentiment analysis model can be extended to other domains, we applied it to tweets concerning Elon Musk's buyout deal of Twitter and to get a glimpse of how the public reacted to the news. We found that the sentiment arc was similar to the pandemic's.

5.2 Future Work

We believe that this tracking system is a good step towards a more advanced and meaningful exploration of news on social media streams. It can facilitate research across various fields, including but not limited to: data science, digital humanities, psychology, and sociology. It can also aide governments and decision-makers monitor the public mood towards entities or trends. We can look into making the system real-time, so that results are delivered results instantly and continuously.

Furthermore, our system automatically detects sentiment changes, but manual analyses have to be conducted to explain the temporal trends. In the future, we plan to explore methods that could automate the reasoning process. We intend to explore the application of aspect-level sentiment analysis instead of document-level for more accurate annotation. The incorporation of a sarcasm detection system might contribute to a better accuracy as well.

Finally, we hope to tune our method to apply it to the Arabic social media. This would require a corpus of labeled Arabic social media posts and modifications in the data cleaning and pre-processing steps.

Appendix

Appendix A

Lists

List of Figures

List of Tables

Bibliography

[1] Hao Wang, Dogan Can, Abe Kazemzadeh, François Bar, and Shrikanth Narayanan. A system for real-time Twitter sentiment analysis of 2012 U.S. presidential election cycle. In *Proceedings of the ACL 2012 System Demonstrations*, pages 115–120, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[2] Peter Kuppens and Philippe Verduyn. Emotion dynamics. *Current Opinion in Psychology*, 17:22–26, 2017. Emotion.

[3] Anastasia Giachanou and Fabio Crestani. Tracking sentiment by time series analysis. SIGIR '16, page 1037–1040, New York, NY, USA, 2016. Association for Computing Machinery.

[4] Svetlana Kiritchenko, Xiaodan Zhu, and Saif Mohammad. Sentiment analysis of short informal text. *The Journal of Artificial Intelligence Research (JAIR)*, 50, 08 2014.

[5] Amir Gandomi and Murtaza Haider. Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, 35(2):137–144, 2015.

[6] Henrico Bertini Brum and Maria das Graças Volpe Nunes. Semi-supervised sentiment annotation of large corpora. In Aline Villavicencio, Viviane Moreira, Alberto Abad, Helena Caseli, Pablo Gamallo, Carlos Ramisch, Hugo Gonçalo Oliveira, and Gustavo Henrique Paetzold, editors, *Computational Processing of the Portuguese Language*, pages 385–395. Springer International Publishing, 2018.

[7] Alaa Mahmood, Siti Kamaruddin, Raed Naser, and Maslinda Mohd Nadzir. A combination of lexicon and machine learning approaches for sentiment analysis on facebook. *Journal of System and Management Sciences*, 10, 11 2020.

[8] Shalak Mendon, Pankaj Dutta, Abhishek Behl, and Stefan Lessmann. A hybrid approach of machine learning and lexicons to sentiment analysis: Enhanced insights from twitter data of natural disasters. *Inf. Syst. Frontiers*, 23:1145–1168, 2021.

[9] Marco Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. pages 97–101, 02 2016.

[10] Christiane Fellbaum. *WordNet: An Electronic Lexical Database.* Bradford Books, 1998.

[11] Abdelwahab A. Abdelkader H. Alqasemi, F. Constructing automatic domain-specific sentiment lexicon using knn search via terms discrimination vectors. *International Journal of Computers and Applications, 41(2)129-139*, 2019.

[12] Venkateswarlu Bonta, Nandhini Kumaresh, and N. Janardhan. A comprehensive study on lexicon based approaches for sentiment analysis. *Asian Journal of Computer Science and Technology*, 8:1–6, 03 2019.

[13] C. Hutto and Eric Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1):216–225, May 2014.

[14] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[15] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[16] Yu Meng, Jiaxin Huang, Guangyuan Wang, Zihan Wang, Chao Zhang, and Jiawei Han. Unsupervised word embedding learning by incorporating local and global contexts. *Frontiers in Big Data*, 3, 2020.

[17] Pandu Nayak. Understanding searches better than ever before, Oct 2019.

[18] amp; Meurer W. J. Tolles, J. Logistic regression: Relating patient characteristics to outcomes. *Jama*, 316, 2016.

[19] Md. Taufiqul Haque Khan Tusar and Md. Touhidul Islam. A comparative study of sentiment analysis using nlp and different machine learning techniques on us airline twitter data. *CoRR*, abs/2110.00859, 2021.

[20] W.P. Ramadhan, S.T.M.T. Astri Novianty, and S.T.M.T. Casi Setianingsih. Sentiment analysis using multinomial logistic regression. In *2017 International Conference on Control, Electronics, Renewable Energy and Communications (ICCREC)*, pages 46–49, 2017.

[21] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning.* MIT Press, 2016. http://www.deeplearningbook.org.

[22] Lei Zhang, Shuai Wang, and Bing Liu. Deep learning for sentiment analysis : A survey, 2018.

[23] Ashkan Eliasy and Justyna Przychodzen. The role of ai in capital structure to enhance corporate funding strategies. *Array*, 6:100017, 07 2020.

[24] Sepp Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 06(02):107–116, 1998.

[25] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Computation*, 31(7):1235–1270, 07 2019.

[26] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[28] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.

[29] Mitchell Lewis Kiley Dilan Danforth Christopher M Dodds Peter Sheridan Reagan, Andrew J. The emotional arcs of stories are dominated by six basic shapes. 5, 2016.

[30] Peter Sheridan Dodds, Kameron Decker Harris, Isabel M. Kloumann, Catherine A. Bliss, and Christopher M. Danforth. Temporal patterns of happiness and information in a global social network: Hedonometrics and twitter. *PLoS ONE*, 6(12):e26752, dec 2011.

[31] W.B. Claster, Phillip Pardo, Malcolm Cooper, and Kayhan Tajeddini. Tourism, travel and tweets: Algorithmic text analysis methodologies in tourism. *Middle East J. of Management*, 1:81 – 99, 01 2013.

[32] Rabindra Lamsal. Design and analysis of a large-scale covid-19 tweets dataset. *Applied Intelligence*, 51(5):2790–2804, 2021.

[33] Furqan Rustam, Madiha Khalid, Waqar Aslam, Vaibhav Rupapara, Arif Mehmood, and Gyu Sang Choi. A performance comparison of supervised machine learning models for covid-19 tweets sentiment analysis. *PLOS ONE*, 16(2):1–23, 02 2021.

[34] Shuo Feng, Huiyu Zhou, and Hongbiao Dong. Using deep neural network with small dataset to predict material defects. *Materials Design*, 162:300–310, 2019.

[35] Megan Doman, Jacob Motley, Hong Qin, Mengjun Xie, and Li Yang. Shifting trends of covid-19 tweet sentiment with respect to voting preferences in the 2020 election year of the united states. 2022.

[36] Social-media and intraday stock returns: The pricing power of sentiment. *Finance Research Letters*, 30:116–123, 2019.

[37] Johan Bollen, Bruno Gonçalves, Ingrid van de Leemput, and Guangchen Ruan. The happiness paradox: your friends are happier than you. *EPJ Data Science*, 6(1):4, May 2017.

[38] Chao Yang and Padmini Srinivasan. Life satisfaction and the pursuit of happiness on twitter. *PLOS ONE*, 11(3):1–30, 03 2016.

[39] Hansen Schwartz, Johannes Eichstaedt, Margaret Kern, Lukasz Dziurzynski, Richard Lucas, Megha Agrawal, Gregory Park, Shrinidhi Lakshmikanth, Sneha Jha, Martin Seligman, and Lyle Ungar. Characterizing geographic variation in well-being using tweets. *Proceedings of the International AAAI Conference on Web and Social Media*, 7(1):583–591, Aug 2021.

[40] Anastasia Giachanou, Ida Mele, and Fabio Crestani. Explaining sentiment spikes in twitter. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, page 2263–2268, New York, NY, USA, 2016. Association for Computing Machinery.

[41] Hyeju Jang, Emily Rempel, David Roth, Giuseppe Carenini, and Naveed Zafar Janjua. Tracking covid-19 discourse on twitter in north america: Infodemiology study using topic modeling and aspect-based sentiment analysis. *J Med Internet Res*, 23(2), Feb 2021.

[42] Househ M Hamdi M Shah Z Abd-Alrazaq A, Alhuwail D. Top concerns of tweeters during the covid-19 pandemic: infoveillance study. *Journal of medical Internet research 22(4):e19016*, 2020.

[43] The GDELT Project. Visualizing twitter's evolution 2012-2020 and how tweeting is changing in the covid-19 era. https://www.forbes.com/sites/kalevleetaru/2019/03/04/visualizing-seven-years-of-twitters-evolution-2012-2018/, 2020.

[44] Lei Zhang and Bing Liu. *Sentiment Analysis and Opinion Mining*, pages 1152–1161. Springer US, Boston, MA, 2017.

[45] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016.

[46] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in bertology: What we know about how BERT works. *CoRR*, abs/2002.12327, 2020.

[47] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147, 2003.

[48] J. Guiñón, Emma Ortega, José García-Antón, and V. Pérez-Herranz. Moving average and savitzki-golay smoothing filters using mathcad. 01 2007.

[49] Jianhua Yin and Jianyong Wang. A dirichlet multinomial mixture model-based approach for short text clustering. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, page 233–242, New York, NY, USA, 2014. Association for Computing Machinery.

[50] Jocelyn Mazarura and Alta de Waal. A comparison of the performance of latent dirichlet allocation and the dirichlet multinomial mixture model on short text. pages 1–6, 11 2016.

[51] Juan M. Banda, Ramya Tekumalla, Guanyu Wang, Jingyuan Yu, Tuo Liu, Yuning Ding, Katya Artemova, Elena Tutubalin, and Gerardo Chowell. A large-scale COVID-19 Twitter chatter dataset for open scientific research - an international collaboration, May 2020.

[52] Francesco Barbieri, José Camacho-Collados, Leonardo Neves, and Luis Espinosa Anke. Tweeteval: Unified benchmark and comparative evaluation for tweet classification. *CoRR*, abs/2010.12421, 2020.

[53] Sara Rosenthal, Noura Farra, and Preslav Nakov. SemEval-2017 task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[54] Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-Collados. Timelms: Diachronic language models from twitter, 2022.

[55] Shaunak Joshi and Deepali Deshpande. Twitter sentiment analysis system. *CoRR*, abs/1807.07752, 2018.

[56] Marco Pota, Mirko Ventura, Rosario Catelli, and Massimo Esposito. An effective bert-based pipeline for twitter sentiment analysis: A case study in italian. *Sensors*, 21(1), 2021.

[57] David C. Blair. *Information Retrieval, 2nd ed. C.J. Van Rijsbergen. London: Butterworths; 1979: 208 pp.*, volume 30. 1979.

[58] Gideon James Rubin, Sarah Harper, Paolo Diaz Williams, Sanna Öström, Samantha Bredbere, Richard Amlôt, and Neil Greenberg. How to support staff deploying on overseas humanitarian work: a qualitative analysis of responder views about

the 2014/15 west african ebola outbreak. *European journal of psychotraumatology*, 7:30933–30933, Nov 2016.

[59] Talita Greyling, Stephanie Rossouw, and Tamanna Adhikari. Happiness-lost: Did governments make the right decisions to combat covid-19? GLO Discussion Paper 556, 2020.

[60] Covid-19, lockdowns and well-being: Evidence from google trends. *Journal of Public Economics*, 193:104346, 2021.

[61] Maike Luhmann, Wilhelm Hofmann, Michael Eid, and Richard E. Lucas. Subjective well-being and adaptation to life events: a meta-analysis. *Journal of personality and social psychology*, 102(3):592–615, Mar 2012.

[62] Giliberto Capano, Michael Howlett, Darryl S L Jarvis, and M Ramesh. Long-term policy impacts of the coronavirus: normalization, adaptation, and acceleration in the post-COVID state. *Policy and Society*, 41(1):1–12, 01 2022.

[63] Niladri Dash. Context and contextual word meaning. *Journal of Theoretical Linguistics*, 5, 01 2008.