Mental Health Analysis of Cancer-Diagnosed Patients With The Lowest Survival Rate and Their Caregivers

by

Irin Hoque Orchi
19301023
Sabrina Tajrin Prarthona
19101397
Nafisa Tabassum
19101184
Jaeemul Hossain
19101632
Iftekhar Alam Ishti
18201040

A Thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
BRAC University
May 2023

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material that has been accepted, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Iftekhar Alam Ishti

Iftekhar Alam Ishti
18201040

Irin Hoque Orchi

Irin Hoque Orchi
19301023

Sabrina T. Prarthona

Sabrina Tajrin Prarthona
19101397

Nafisa Tabassum

Nafisa Tabassum
19101184

Jaeemul Hossain

Jaeemul Hossain
19101632

Approval

The thesis/project titled "Mental Health Analysis of Cancer-Diagnosed Patients With The Lowest Survival Rate and Their Caregivers" submitted by

1. Iftekhar Alam Ishti (18201040)
2. Irin Hoque Orchi (19301023)
3. Sabrina Tajrin Prarthona (19101397)
4. Nafisa Tabassum (19101184)
5. Jaeemul Hossain (19101632)

Of Spring, 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May 22, 2023.

Examining Committee:

Supervisor:
(Member)

Farig Yousuf Sadeque

Dr. Farig Sadeque
Assistant Professor
Department of Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
BRAC University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
BRAC University

Abstract

Throughout the world, millions of people and their families are impacted by the serious illness of cancer. The total number of new cancer cases worldwide in 2020 was predicted to reach 18.1 million. Serious emotional disorders, like depression, are often present in cancer patients due to many factors, including the intensity of certain situations, the negative consequences of their long treatment, or the deaths of other cancer patients. Therefore, keeping an eye on the patient's moods is crucial to their ongoing treatment. Many cancer patients use online social media sites such as Facebook and Twitter to communicate their thoughts and emotions about their treatments, as well as the difficulties associated with them, in the form of posts or messages. From these sources, we can get good information about the mood of those patients, which will further help us with their treatment. After applying the necessary pre-processing to this data, we can apply sentimental analysis methods, which will help us predict the positive or negative emotions of cancer patients on these online platforms. We can give better psychological support to these patients after analyzing their mental health. So, our objective is to design a model capable of identifying such actions, as among all the cancers we are working on, five have the lowest survival percentage.

Keywords: Sentimental Analysis, Psychological Support, Cancer Patient, SBERT, RNN, GRU, LSTM, Few Shots, Emotional Valence, and Arousal.

Dedication

This paper is dedicated to individuals who have been diagnosed with cancer and continue to fight through the challenges of the disease with unwavering determination.

Acknowledgement

Firstly, we express our gratitude to the Almighty, because of whom our thesis was completed without any significant disruptions. Then we would like to extend our gratitude to our supervisor, Dr. Farig Yousuf Sadeque Sir, for his invaluable guidance, assistance, and motivation throughout the research and thesis preparation phases. In the end, we wish to convey our appreciation to our beloved parents and friends for their unwavering assistance and incomparable understanding throughout the duration of our research.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

$CBOW$ Continuous Bag Of Words

GRU Gated Recurrent Unit

KNN K- Nearest Neighbour

NLP Natural Language Processing

RNN Recurrent Neural Network

$SBERT$ Sentence Bidirectional Encoder Representations from Transformers

SVM Support Vector Machine

Chapter 1

Introduction

1.1 What is Emotion Detection?

Recognizing a person's emotional state, such as rage, perplexity, or dishonesty, across voice and nonvoice mediums is the process of mood detection. The most popular method examines the speech signal's properties and any accessible word used as an extra input. Additionally, mood detection is the precise determination of an individual's emotional state based on surrounding data. When a better understanding of a text's underlying tone is required, we often discover that emotion analysis has a substantial impact on business choices. This technology has gained popularity as a result of enhanced text analysis using NLP, machine learning, and computational linguistics to extract emotion and satisfaction significance.

1.2 Significance of Using Emotion Detection

Emotions play a vital role in the communication between consumers of various cultural languages during digital interactions. In general, there are six sorts of emotions: delight, astonishment, disgust, rage, melancholy, and dread. The communication of emotions via plain text, texts, tweets, and the like is generally casual and unstructured. Consequently, the requirement to expand our perspectives to recognize emotions from such text data has expanded dramatically. Detecting complicated emotions from enormous data volumes has become a severe difficulty for anyone working with data. Emotion detection identifies and depicts correct conclusions and emotions by using sophisticated computer approaches to effectively evaluate internet information.

1.3 Methods of Mental Health Recognition

There really are four distinct strategies for text-based mood classification, listed below:

- Keyword Spotting Method
- Lexical Affinity Procedure
- Learning-Based Approach

- Hybrid Based Approach

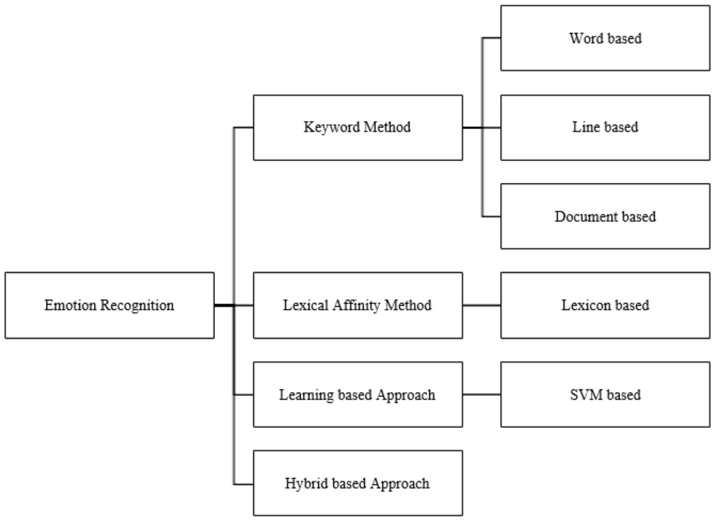

Figure 1.1: Emotion Detection Methods[20]

1.4 Analyzing Mental Health using NLP

Natural language processing (NLP) is a computer-aided method for text analysis based on a collection of concepts and a collection of technologies. It focuses on the interconnections between computers and (natural) human languages. NLP presents texts derived from natural language at one or more levels of lingual analysis to attain human-like language processing for a number of activities or tasks.

NLP has two techniques for analysis. They are:

1. Keyword Analysis or Pattern Matching Technique
2. Syntactic-driven Parsing Technique.

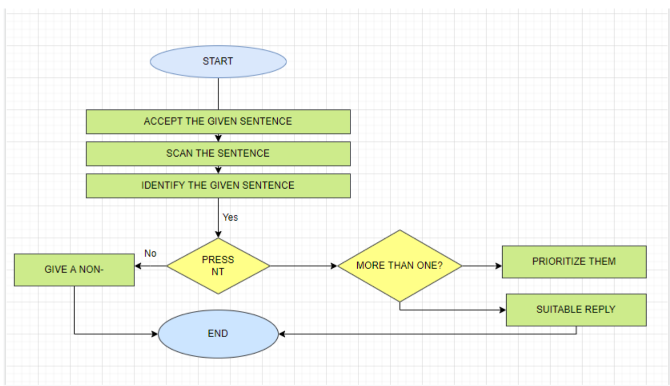

Figure 1.2: Flow Chart for Keyword Analysis

From Figure 1.4, we can see that the lowest accuracy rate is 88.40% and the highest rate is 99.08% for mood detection. So, we can say that by using Natural Language Processing (NLP), we can get a successful accuracy rate for detecting human mood.

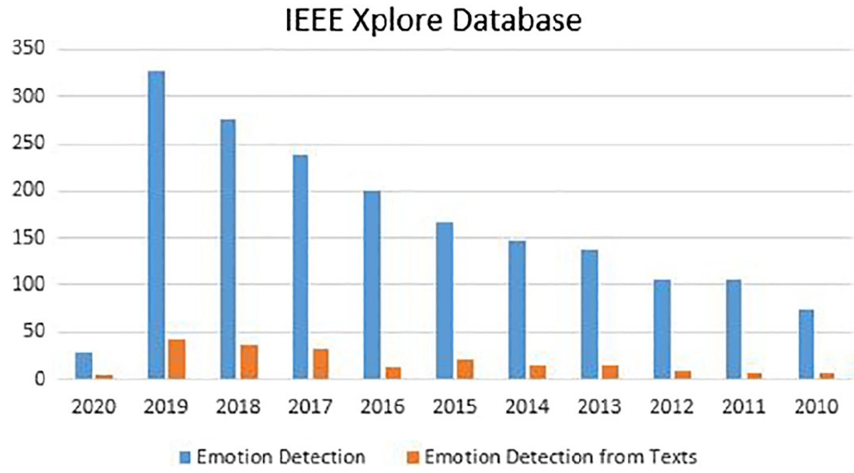

Figure 1.3: Research in text-based mental health detection[9]

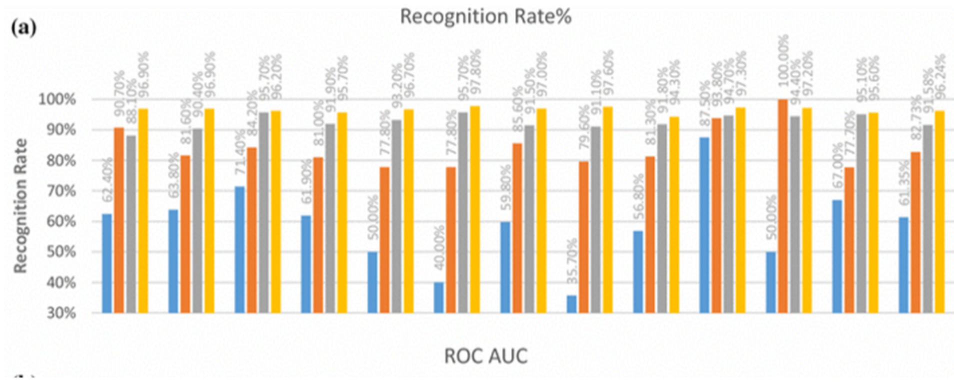

Figure 1.4: Recognition rate for emotion detection[16]

1.5 Problem Statement

The research is to address: "The problem is to identify the mental health of cancer patients with the lowest survival percentage using natural language processing techniques so that we can get an idea of their mental health"

1.5.1 Description of the problem

We all know that cancer treatment is a lengthy and arduous process. Everyone around them is focused on improving their physical health via chemotherapy, radiation therapy, and a variety of other cancer-fighting therapies. But we are quite worried about what is going on in their heads during this depressed period of therapy. Determining the state of their mental health is also an important aspect of their therapy. So we aim to forecast the emotions of cancer patients with the lowest survival rates using data from social media, which will help us take better care of these patients. According to research conducted in the UK, the following cancers have survival rates below 60%: brain cancer (12.8%), liver cancer (13%), lung (17.6%), leukemia (55.4%), and colon (57.7%) [17]. We collected our text data from three main social platforms, which are Reddit, Health Board, and Daily Strength, and further analyzed and annotated our data by arranging it with its valence and arousal states.

1.6 Research Objective

Our objective for the paper is to

- Construct a natural language processing (NLP)-based method for assessing the mental health of cancer patients with the lowest chance of survival.
- Classify the emotions according to their mental states.
- Maximize the precision and accuracy of the detection.
- Concentrate on faster prediction, resulting in greater time efficiency.
- Publish and disseminate our manuscript to computer science students so that it may be useful for their future research.

Chapter 2

Literature Review

The wide and comprehensive research on mood detection conducted by researchers from across the world is vast and exhaustive. Multiple researchers all around the globe are giving emphasis to developing a more precise technique for recognizing the emotional state of a human being. Basically, we will try to find out if a brain cancer patient is suffering from depression, anxiety, illness, etc. Moreover, we will do it text-based by extracting and predicting analyses of the reviews.

MATLAB 2016b, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Decision Tree (DT), and Ensembles were used in the work [3]. First, the authors collected data about depression-related postings through Facebook and other social media sites. They used NCapture to gather data from Facebook since it was impossible to determine whether or not Facebook users had posted comments. However, they evaluated the effectiveness of Facebook comment detection using the aforementioned categorization techniques. According to the results of the investigation, Decision Tree is the most effective model overall. KNN delivers the maximum accuracy, while Decision Tree offers the highest recall and F-measure for the category of depression-related comments from Facebook users. The decision tree delivers the greatest outcomes in terms of accuracy, recall, and F-measure, just as it does with linguistic style. The investigation achieved between 60 and 80 percent accuracy.

The authors of the research [8] intended to identify depression-related messages on the social media site Reddit. They used machine learning and Natural Language Processing (NLP) techniques to train the data and evaluate the efficacy of their proposed approach. In addition, they identify a lexicon that is more prominent in depressive tales. The results reveal that the performance precision of their proposed technique may be significantly enhanced. With an accuracy of 80% and 0.80 F1 scores, respectively, Bigram and the Support Vector Machine (SVM) classifier are the best single features for detecting depression. With 91% accuracy and 0.93 F1 scores, the Multilayer Perceptron (MLP) classifier performs best in identifying depression, highlighting the strength and usefulness of the combined features (LIWC+LDA+bigram).Their study reveals that judicious feature choices and their varied feature combinations may lead to greater performance gains.

The research [12] focuses on predicting a person's emotional state based on shared postings and language from social media sites. Various posts, messages, and photographs from Facebook, Instagram, and other platforms were gathered for this purpose. However, they primarily focused on employing a Twitter dataset to measure people's sadness from their postings. For this objective, many classifiers are utilized, including the decision tree technique, the random forest approach, the assist vector machine method, the naive Bayes approach, and a suggested hybrid algorithm. It is found that the suggested hybrid method, which is a fusion of assist vector machines and naive Bayes, outperforms the separate classifiers in terms of accuracy. More variation factors are included in the suggested method to fine-tune the performance measure.

Another fantastic piece of research [2] investigates the sentimental analysis of data from social networking platforms, which is able to decipher the underlying emotional undercurrent of a string of words or texts. The purpose of this analysis is to determine whether the tone of the text is positive, negative, or neutral in order to gain additional insight into how well we can anticipate a person's level of depression. They primarily focused their analysis on the Twitter dataset and the 20 newsgroup datasets, both of which were used to assess the effectiveness of their methods. In the process of preprocessing the data, a number of steps were carried out, including tokenization, the elimination of stop words, and stemming. Approaches based on meta-learning or voting can be utilized in order to train the dataset. This strategy utilizes multiple classifiers and selects the resulting label based on which one receives the most votes. The nested voting operator employs a majority rule for classification and regression, on top of the predictions made by the inner learners. Classification methods such as the support vector machine (SVM), the Naive Bayes algorithm (NB), and maximum entropy were used for the internal learners (ME). Results show that the Support Vector Machine (SVM) outperforms the Naive Bayes and Maximum Entropy classifiers. The results produced using SVM are 91% accurate, those from Naive Bayes are 83% accurate, and those from Maximum Entropy are 80% accurate.

Moreover, in the study [1], the authors used Twitter feeds to analyze user emotions with a focus on sadness. For analyzing emotions, they mostly employed NLP and machine learning methods. In addition, they enhanced the efficiency of their suggested method. Through the social media site Twitter, the data is being analyzed. This work employs the Naive-Bayes and Support Vector Machines classifiers in order to detect tweets that exhibit indications of mental discomfort and sadness. SVM was outperformed by Multinomial Naive Bayes, which had an F1 score of 79.73 as opposed to 83.29 for SVM. According to their analysis, the count of false positives may be minimized by appending a level of expert recommendations to the model. This would boost the accuracy of recognizing depression using sentiment analysis.

Following Paper [6], Their primary objective is to develop efficient methods for recognizing emotions from diverse sources, including text and voice. In this paper, they examined the research done on categorizing emotions based on text and voice. In their research to improve accuracy, they have applied several word-based and sentence-based methodologies, machine learning, natural language processing techniques, etc. The study compared the efficacy of keyword-based and machine-learning algorithms for recognizing emotions to see which method was more effective for various types of datasets. This study utilizes nonlinear and linear classifiers, such as GMM, SVM, and HMM, to categorize utterances into feelings with the purpose of finding those that exhibit indicators of mental distress and sadness. In addition, hybrid and ensemble systems are applied to achieve superior outcomes. Hybrid systems tend to perform better than systems that rely solely on a single information source. It appears that individual classifiers perform less effectively than ensemble systems. They are working on developing a model that can evaluate multiple emotions in a single message and comprehend the motivation behind a feeling.

In another noteworthy study[10], the major goal of their research is to create an application that scans Twitter users' postings to diagnose personal sadness using machine learning. This program informs the user about their previous emotional state based on their present feelings and assists them in taking appropriate action. They collected data from Twitter postings using Python and JSON. The system used Naive Bayes as a machine learning method, NLP as a Lexicon-based method, and Feed Forward neural network as a Deep Learning method. They also employed the SVM classifier in conjunction with the Naive Bayes to close the gap between positive and negative accuracy. After retrieving the database results, the system will assess the tweets in three models, delivering anticipated sentiments as Neutral, Negative, or Positive. The system will use the positive anticipated expected average sentiment if it returns two positive outcomes and one neutral or negative output, just as it would if two negative outcomes and two neutral outcomes were returned. After processing 10,882 tweets from Twitter, containing both English and Malay tweets, the Naive Bayes algorithm achieved an accuracy of 0.68, whereas the NLP achieved an accuracy of 0.64 since they could only deal with English tweets. Feed-forward Neural Network, on the other contrary, attained a precision of 0.72 since it could deal with both English and Malay tweets.

According to a different paper [5], both unsupervised and supervised approaches were used to identify the sentiments of tweets and short message services, respectively. Unsupervised approaches were applied to various datasets of Twitter posts, while supervised approaches were applied to datasets of text messages or SMS. Essentially, positive, negative, and neutral data were utilized as outcomes to determine the moods of the datasets. Various machine learning methods, including Support Vector Machines, Maximum Entropy, and Multinomial Naive Bayes (MNB) were used to identify the sentiment of tweets and evaluate the efficacy of different feature combinations. In this study, the unsupervised strategy achieved an accuracy of 80.68 percent, whereas the supervised or lexicon-based approach achieved an accuracy of 75.20 percent.

In accordance with the paper [4], this method of diagnosing human depression using facial characteristics was developed. Following is a study in which the authors provide a method for recognizing faces, and demonstrate how this may be used to infer information about a person's emotional state. They employed image processing methods, which can be performed on a real-time basis or in the format of VDO files from CCTV cameras. This data may be obtained from the cameras. To begin, they separated the aspects of the psychological state into the three categories of learning, emotions, and patterns of behavioral expression. These are the categories. They have categorized the several ways in which people convey their emotions, including facial expressions, gestures, and noises. A study on psychoeducational literacy communication found that voice communication had only a 7% emotional impact, while voice communication had a 38% emotional impact. This is in comparison to the 55% emotional impact that facial expressions have, indicating that human facial expressions are the most important factor in identifying emotions.

The study in [7] centers on emotion detection and prediction exploiting users' daily behaviors. According to S. Pouneh, Z. Min, and M. Shaoping (2018), studies have shown that our daily activities and mood states are influenced by our mood states, but this isn't the only method. Our daily activities have an impact on our mood as well. In this study, they employed "lifelong" user data to identify and forecast users' moods. The emotional states presented in this essay are grounded in Thayer's two-dimensional theory of mood. This study is the first to thoroughly examine the physical information gleaned from lifelog and how it relates to factors that influence mood, such as biometrics, physical activity, sleep patterns, food, and the user's surroundings. Their study shows that the links exist and are significant. Lifelogging is the practice of digitally documenting a person's everyday activities for a number of reasons and in great detail. Such information can be maintained to track daily activities and enhance the quality of the human experience. This lifelogging technology has already been used to recollect human memories and forecast and diagnose health risks. Utilizing mobile applications and wearable sensors, the lifelog data is automatically logged. The first stages include data extraction, preparation, and preprocessing. Some of them are data shifting based on sleep and waking time, summarization, and feature normalization. Then, they acquired a number of characteristics and classified them into six sections, including biometrics, actions, proximity and surroundings, dietary habits, consumer history records, and sleep statistics. Every morning, life loggers manually identify the data according to Thayer's two degrees of mood. The primary objective was to predict the user's mood in 2 dimensions which are valence and arousal detecting the user's mood in two dimensions using lifelog data from today. For the sake of coherence following data extraction, we shifted the data in two directions which are one for emotion prediction and emotion detection. They reasoned that the more time passes between an activity or biometric collection and going to bed, the less of an impact it will have on one's mood the following morning. There are better indicators of the mood label provided by the user in the case of mood detection, where the actions and documented biometric identification are relative to the wake-up time and thus labeling. Then the feature was normalized. Moreover, they labeled features of multiple attributes according to a table which includes Sleep Data, Activities, Surroundings, etc. They used Adaptive Boosting in conjunction with SVM and C4.5 decision tree generators to train the models. Due to data limitations, they did not use any deep learning algorithms. For this study, lifelog data from one of NTCIR13's lifelog datasets and data collected on five volunteer life loggers were used. The users hail from various nations. Five users including user 1 from the NTCIR13 dataset contribute 25 days of data, and user 6 contributes 175 days of data, for a grand total of 300 instances that have been labeled. The data contains biometrics, activities, User's Environment, Diet, Sleep, and daily mood labels. The following accuracy was observed in SVM, it was around 82.53% and for C4.5 it was around 81.04%. The challenge of the research was that there was no adequate data for research, and since the users are from different places, the data varies according to their surroundings. However, they want to overcome the challenges in the future through more data and also increase the number of parameters accordingly.

Chapter 3

The Dataset

3.1 Dataset Type

The objective of our study is to conduct sentiment analysis on the mental health of individuals diagnosed with cancer. The extracted data from the aforementioned posts are in the form of textual content. The individual seeks to communicate their mental state through various social media platforms, in order to seek some mental help or suggestions. Therefore, we could gather approximately ten thousand textual data. So, text dataset will be our area of focus.

3.2 Dataset Collection

For our study, we need the postings of the patients, from which we will analyze the emotions associated with them; thus, we scraped the posts of the patients and their caregivers mostly from Reddit, Daily Strength, and the Health Board. These are some of the forums where patients, together with their friends, family members, or caregivers, communicate their ideas, thoughts, pains, and other emotions they experience throughout their cancer journey. Our first goal was to find out which types of cancer, such as brain, colon, liver, lung, and leukemia, had the lowest survival rates. Consequently, we obtained unprocessed data from the comments of individuals diagnosed with these malignancies, and we can guarantee that we did not rely on datasets from other resources.

3.3 Data Analysis

We gathered about 10,087 posts from the aforementioned social forums. Further analysis revealed that there were 875 cases of brain cancer, 2122 scenarios of colon cancer, 3083 occurrences of liver cancer, 1054 instances of leukemia, and 2952 lung cancer cases.
After reviewing all the entries in our dataset, we determined that the greatest length of a post is around 613 words, while the shortest length is just 3 words. Therefore, the average post length is 308 words. In addition, we divided the feelings of patients gleaned from their postings into three primary categories: positive, negative, and neutral. When we recognized a negative emotion in a post, we assigned it a number

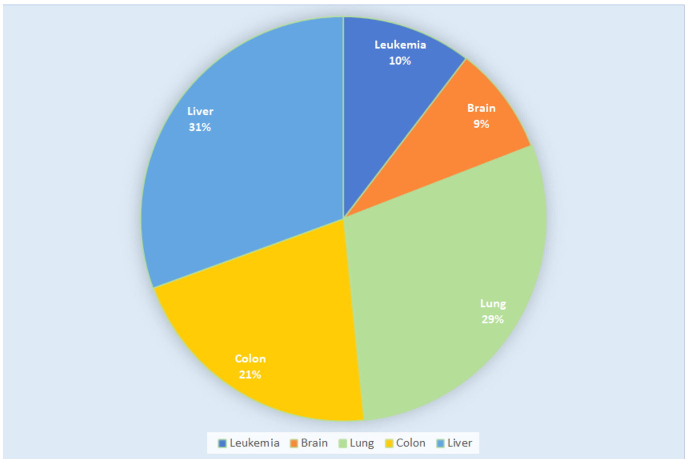

Figure 3.1: Percentage of Comments Collected from Reddit, Health Board, and Daily Strength

of -1 or -2 depending on its intensity, and when we found a positive sentiment, we assigned it a score of 1. Negative emotions include sorrow, despair, worry, and concern, while positive emotions include hope, curiosity, offering guidance, and being joyful or content. Posts that looked to be advertisements/needed advice/confused/others were categorized as neutral and assigned a score of 0.
A total of 2,373 data were manually annotated from a sample of 10,087 posts. The posts were categorized based on their respective conveyed emotional states, namely positivity, negativity, or neutrality. The positive posts were assigned a score of 1, whereas the negative posts were assigned a score of -1 or -2, depending on the degree of severity. Posts that conveyed extreme sadness or depression were assigned a score of -2, while negative posts that only conveyed anxiety or concern were assigned a score of -1. The following is a data representation consisting of 500 scored data out of a total of 2373 annotated data.

3.4 Data Representation

In this study, we collected raw data for analysis from three different platforms. After that, we had to combine all these data for getting a proper representation. These data were initially classified into two divisions: 1) Arousal and 2) Valence.

3.4.1 Arousal, Valence

Arousal: Arousal refers to a state of physical and mental wakefulness or the activation of sensory organs to the point of perception. It includes turning on the brain's "ascending reticular activating system" (ARAS), which regulates the sleep-wake cycle, the endocrine system, and the motor nervous system. This results in elevated heart and blood pressure as well as a sensory state of acuity, desire, agility, and reaction. Some examples of arousal could be happiness, sadness, anger, etc.

Valence: The emotive quality of valence, also known as hedonic tone, refers to the inherent attractiveness or goodness, or positive or negative valence, of a situation, an item, or an event. An example of emotions that indicate "negative," such as

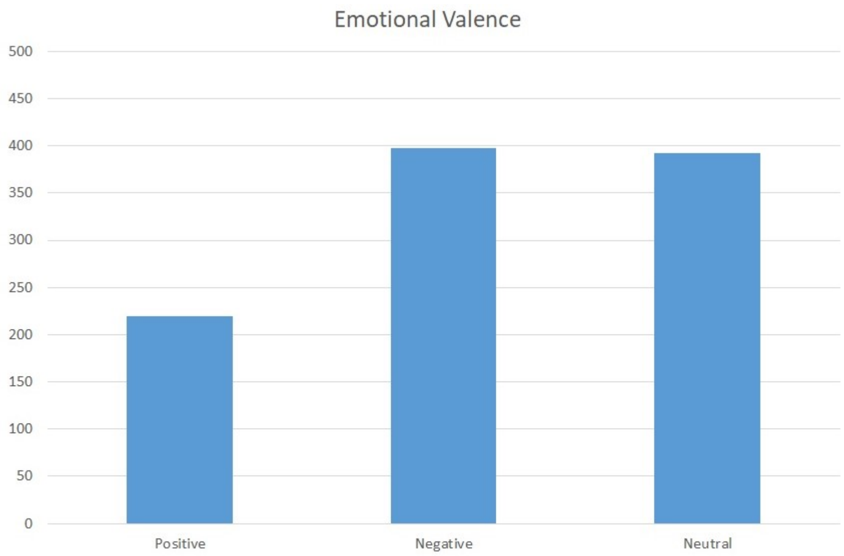

Figure 3.2: Emotional Valence of our scored data

anger and fear, have a negative valence, and "positive" emotions could be happiness and laughter.
For the pre-analysis part, we arranged the data in their respective arousal and valence states. For the (Arousal: Valence) representation, we chose:

Valence	**Arousal**
Very negative	-2
negative	-1
neutral	0
positive	1

Table 3.1: Arousal/Valence Relation

In the above, arousal indicates the intensity of that particular emotion. The value (-2, -1, 0, 1) in the column "Valence" means (very negative, somewhat negative, neutral, somewhat positive, or very positive). A graphical representation of how the data were arranged in an Excel file is shown below:
In this manner, we arranged posts related to each of the cancer diseases that have the lowest survival rate. After that, each member of our group scored the arousal rate of the first 200 posts for each cancer disease individually. After anonymous scoring by each of our members, we calculated the number of negative, neutral, and positive scored posts separately for each cancer disease. After scoring and calculating, we found the following information:
The total numbers of positive, negative, and neutral scores are 219, 398, and 392, respectively. From the above analysis of the score, we were able to get an average of around 44 positives, 80 negatives, and 78 neutral emotional valences.

Posts	Valence	Arousal
i should of come here early , where to start my other family members my mom,dad,two sisters are all gone , one sister murdered, mom died from cancer, dad brain tumor , one sister just a few month ago i feel lonely i miss my mom so bad mom was only 58 ,dad 65,one sister 35, the other 63 what on why am i still here life been kind of cruel to me , especially if i get stressed out then it bothers me . I'm fed up	Very negative	-2
i had posted in here days ago regarding my cousin's husband. Her husband's cancer is getting better by chemo therapy.feeling good after hearing from my cousin as he's getting positive response from the treatment	Positive	1
I have gone in to a few of my groups trying to round up prayer for my cousin.Her husband was diagnosed with stage 4 brain cancer.he is leaving for cancer centers of america tomorrow.sadly,last night he developed pancreatitis and had to be put back in hospital..very worried about ability to fly tomorrow..please pray for him..his name is John,my cousin michelle. God bless you all.	Neutral	0
My friend Rita is in her 80s and she is frail....Her husband just passed away...a week ago they were told her husband had a brain tumor and cancer...I'm really concerned for her.	Negative	-1

Figure 3.3: Sample Data Representation

Cancer type/disease	**Positive**	**Negative**	**Neutral**
Brain	32	131	37
Lung	33	83	84
Leukemia	37	73	94
Liver	48	84	72
Colon	69	27	105

Table 3.2: Scored results of first 200 data of each cancer patient

3.5 Data Annotation

For our further work, we labeled the data as follows:

- **Sad/Depressed:** The posts which expressed suffering, pain, frustrations, or which seemed like someone was venting out their grievances
- **Anxiety/Concerned:** It consists of the posts which indicate that a patient or caregiver is in panic or tense for some physical condition
- **Hope/Curious to know/Giving Advice:** The posts which conveyed some hope or aim to people or if someone is just curious about something or if anybody is giving a piece of advice to the people
- **Happy/Satisfied:** In most cases, when someone successfully fought against cancer or made some progress in their cancer journey, the posts depicted a happy/satisfied emotion

3.5.1 Cohen's Kappa Score

The Kappa score is a metric utilized to quantify the degree of agreement or disagreement between two human reviewers when assessing subjects. This method is utilized to assess the efficacy of a machine learning classification model by taking into account the level of agreement between two raters in terms of perfect agreement and chance agreement. In our study, the posts pertaining to a particular type of cancer were evaluated through pair scoring by two members of our research team. Both individuals obtained scores in accordance with our established scoring system. Subsequently, we assessed the level of concordance or discordance among them through the utilization of Kappa's coefficient. The following scores were obtained through the implementation of Cohen's Kappa:

Raters	Cohen's Kappa
Rater (1,2)	0.594
Rater (1,3)	0.341
Rater (2,5)	0.207
Rater (3,4)	0.503
Rater (4,5)	0.556

Figure 3.4: Kappa's Score Between Two Raters

A Kappa coefficient ranging from 0.4 to 0.6 denotes a moderate level of concordance between two assessors, whereas a coefficient lower than this range indicates a slight level of concordance. Based on the aforementioned scoring, it is evident that there exists a moderate level of agreement among the pairs of raters (1,2), (3,4), and (4,5). However, the level of agreement is considerably lower between the pairs of raters (2,5) and (1,3). To address this matter, it is necessary to enlist the services of a third reviewer who is commonly referred to as a meta-reviewer. The meta reviewer will assign a single score to a post in the event of disagreement between two raters, by selecting one of the scores provided by the raters.

3.5.2 Meta review

The collected posts consisted of five categories of cancer in total. Each category of cancer post was annotated by two of our members. The posts were gone through and scored between the numbers -2 to 1 that is -2,-1,0, and 1. Based on the feelings it portrayed, these scores were assigned. According to the intensity, entries that appeared to be expressing grief, suffering, or depression received a score of -1 or -2. On the other side, comments that expressed some happy emotions were rated positively and given a score of 1. Positive emotions include feelings of relief following a cancer battle or a sense of accomplishment after making progress in cancer therapy. Some messages expressed no emotion at all; instead, they were merely questions, medication names, or strange advertisements. These comments received a score of 0 and were classified as neutral.

After the pair scoring of specific cancer posts was done by two members, then the same cancer posts were reviewed by a third member of our group. After reading each message, the third person scored it by choosing a number between the prior scores of the first two members. He or she will give this score based on which of the previous two scores seemed more accurate based on the emotions conveyed by the post. In this manner, all five categories of cancer posts were meta-reviewed by our members. 2373 posts were hand-annotated as well as meta-reviewed.

3.6 Data Preprocessing

We need to prepare and filter the raw data for further analysis and the running of the classifiers. Because the data was accumulated from social media postings, there will be multiple superfluous terms in this data. So, first, we lowercase all of the phrases, and then we tokenize all of the sentences by normalizing texts and managing punctuation. After that, we eliminate stopwords, and our data is ready to be processed for future work. Then we utilize Fasttext and TF-IDF as pre-processing techniques before passing data to the classifiers for further processing.

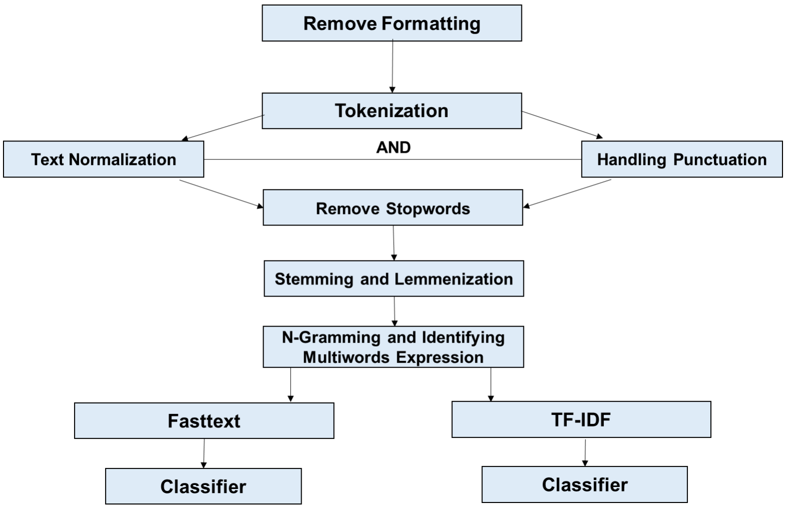

Figure 3.5: Data Pre-Processing and Procedure

The following pre-processing has been done within our dataset so that machine learning models can be implemented through it:

- Several posts contained personal email addresses of individuals. These data are considered to be irrelevant and do not contribute to the achievement of our mission. Consequently, all email correspondences were expunged to maintain the cleanliness and efficiency of our system by eliminating extraneous data
- Within the compiled posts, we encountered specific HTML tags. The aforementioned tags are often regarded as extraneous data as they do not effectively communicate any emotional states. As a component of the data cleansing procedure, it is necessary to eliminate the tags
- To ensure uniformity in the presentation of characters, all textual content of the dataset's posts has been converted to lowercase
- The posts included extraneous characters that were not pertinent to our predictive objective. The special characters were eliminated from the posts to preserve only the significant textual content
- For data cleansing purposes, several accented characters that were present were eliminated
- All the verbs with past, continuous, or any other forms were transformed to their base form for ease of implementation of the models

3.7 Pre-processing Techniques

3.7.1 Fasttext

Word embedding is the process of converting the meaning of a word or sentence into a vector that may then be used in mathematical computations. It creates vector representations of words with connected meanings. It also helps the model better represent the grammatical meaning of the word. To make predictions, we must use machine learning models on the postings we have gathered. Therefore, in order to make these posts machine-readable, we need to transform them into numbers. Although there are several approaches to word embedding, fasttext word embedding works best with our challenge and dataset.

Fasttext is one of the word embedding techniques that enhances the depiction of morphologically rich language. The unit on which fast text provides embeddings is character N-grams instead of words, representing words as the average of these embeddings. Here N is a hyperparameter that usually ranges from 2-6. In this process, we are breaking apart a word by taking N characters at a time. After the character N-gram representation of a word is accomplished, then the skip-gram model is implemented to learn the word embeddings. As the internal framework of the word is not taken into account, it is considered to be a bag of words model featuring a sliding window over a word. It is an extension of the word2vec model. So, its architecture is similar to that of word2vec, which uses CBOW and Skip Gram to determine the vectors.

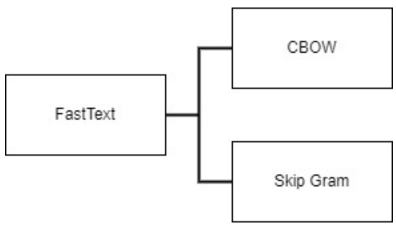

Figure 3.6: FastText components

3.7.2 CBOW(Continuous Bag of Words)

When given the sentence's context as input, a continuous bag of words (CBOW) predicts the target words. By expressing the words in a numerical vector space during the prediction process, it learns the word embeddings. Three layers, including an input layer, an output layer, and a hidden layer, make up the continuous bag of words' basic architecture. The context of the word is provided as input to the input layer. The input vectors are the representation of one hot encoded vector, where each element of the vector represents a distinct word. The target word is finally predicted by the output layer once the hidden layer, a dense layer that learns word embeddings from contexts, has learned all the word embeddings.

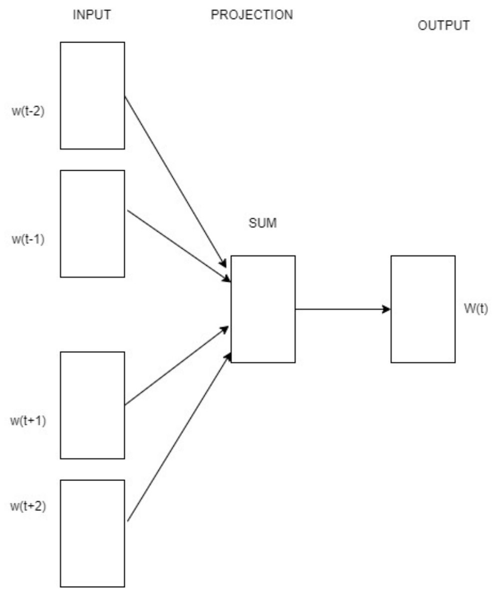

Figure 3.7: CBOW Architecture

3.7.3 Skip-Gram

In contrast to the CBOW architecture, skip-gram outputs the context of the target word when the target word is given as input. It is one of the techniques used in unsupervised learning to identify the words that are most related to a given word. It is a bit more difficult compared to CBOW as multiple target words are to be determined concerning a single target word. Similar to CBOW, it also has three layers, an input layer, a hidden layer, and an output layer. The target word is given as input in the input layer. The dot product between the input vector and the weight matrix is conducted in the hidden layer with no application of the activation function. Then the output layer is given the dot product of the hidden layer. The output layer further calculates the dot product between the hidden layer's output and the weight matrix of the output layer. Finally, the most probable context words are determined by applying the softmax function.

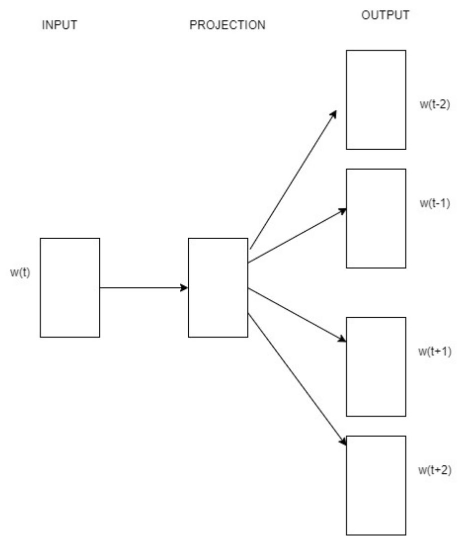

Figure 3.8: Skip-Gram

3.7.4 Advantages of using FastText

Some of the advantages of fasttext that we implemented on our dataset are:

- It captures more fine level and more granular information of the words which can't be acquired through typical embedding processes like Word2vec, and Gloves
- Fasttext has the capability to address the issue of Out of Vocabulary Problems, commonly referred to as OOV problems. The issue of out-of-vocabulary arises when certain lexical items emerge that were not included in the training corpus and are therefore absent from the lexicon of the model. Within the context of word2vec, words that were not included in the training phase are designated as out-of-vocabulary (OOV) words. The out-of-vocabulary (OOV) issue is addressed to a significant degree in Fasttext. The model can promptly recognize an unknown word by leveraging its prior exposure to small portions of the word through other words in the vocabulary. This eliminates the challenge of encountering out-of-vocabulary issues, a prevalent concern encountered by alternative text embeddings
- Fasttext is our first choice because we want to train a custom embedding for our domain. Our dataset contains many medical-specific or other scientific terms related to cancer. It becomes difficult to train such models with standard models like Bert or word2vec. In this case, we have to train our own custom model and the best choice is to implement fasttext embedding
- Fasttext is lightweight. So, we need fewer resources on it and we get a solid baseline in our first attempt. Therefore, it is worth the effort to train the model with fasttext rather than another accessible embedding method

3.8 TF-IDF

TF-IDF: The process of vectorizing text known as Term Frequency-Inverse Term Frequency (TF-IDF) is widely used in the domains of Natural Language Processing and Information Retrieval. The TF-IDF method enables the establishment of a correlation between individual words within a given document and a numerical value that denotes the significance of each word in that particular document. This approach involves ascertaining the significance of a term in a given document in relation to a corpus or set of documents. Each term within a document is converted into numerical values to determine its significance. The TF-IDF methodology confers a numerical value to a given term or converts it into a vector by computing the product of the term's term frequency (TF) and the inverse document frequency (IDF).

Term Frequency(TF): The ratio of the number of times a particular word appears in a text to the total number of words in the document is referred to as the the term frequency of that particular word.

$$TF = \frac{Number\ of\ times\ term\ t\ appears\ in\ a\ document}{Total\ Number\ of\ words\ in\ a\ document} ....(i)$$

Inverse Term Frequency(IDF): The IDF of a term is the proportion of documents in the corpus that contain the term. Through this calculation, words that are unique or contains in a smaller percentage receives higher importance whereas the words which occur more frequently receive lower importance.

$$IDF = log(\frac{Number\ of\ documents}{Number\ of\ documents\ with\ term\ t\ in\ it})........(ii)$$

The TF-IDF value of a word is calculated by multiplying its TF and IDF value.

$$TF - IDF = TF * IDF........(iii)$$

As an illustration, consider the following three sentences: sentence 1, "Beautiful nature"; sentence 2, "Beautiful scenery"; and sentence 3, "Nature scenario beautiful." In order to determine the Term Frequency (TF) of the words "Beautiful," "Nature," and "Scenery" within three given sentences, the frequency of each word will be calculated by dividing the number of times it appears in each sentence by the total number of words in that sentence. Regarding the IDF, a logarithmic function is employed to calculate the ratio of the total number of sentences to the number of sentences that contain the specific word. In this particular instance, the total number of sentences is 3. Regarding the sentiment analysis of the text, the logarithmic function of the ratio (3/3) will be applied to the category "Beautiful", as there are three sentences in total and the term "good" appears in all of them. For the ultimate result, the TF will be multiplied by the IDF.

Chapter 4

Methodology

4.1 Gated Recurrent Unit (GRU)

The Gated Recurrent Unit was created as a solution to the problem of vanishing gradients in conventional recurrent neural networks. Given their structural similarities to LSTMs, GRUs can be thought of as a form of LSTM and even achieve similar results in certain circumstances. When an ordinary RNN's gradient stops decreasing, the update gate and reset gate must be operated by GRU to restore the lost information. The information that will be transmitted to the output is largely determined by these two vectors. They are special because they can be trained to remember details from the past without the effects of time passing on the data, as well as to extract details that are irrelevant to the estimation.

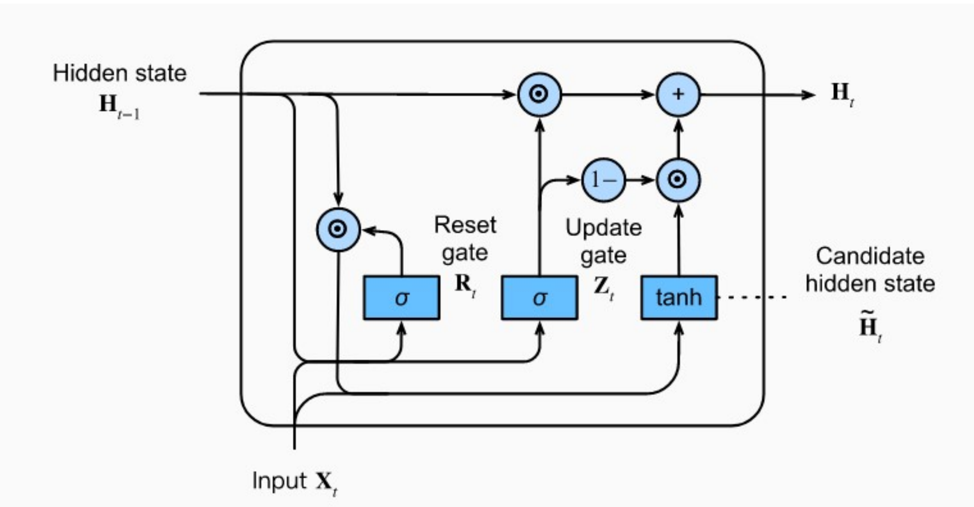

Figure 4.1: Workflow of GRU[15]

Update gates, reset gates, and current memory gates are the three distinct varieties of gates used in GRU. The amount of data from the past that must be used to make predictions is set by the update gate. However, the model employs the reset gate to determine what, if any, of the previous data to forget. The current memory gate is often disregarded during discussions on Gated Recurrent Unit Networks. In the same way that the Input Modulation Gate is a part of the Input Gate, the Zero-mean Input Gate is a part of the Reset Gate. This makes the input nonlinear. First, calculate the update gate (z_t) for the time step.

$$z_t = \sigma\left(w^{(z)}x_t + v^{(z)}h_{t-1}\right) \text{........}(i)$$

The input vector for this network node process is denoted by x_t. Their parameter weight matrices (W_z) are multiplied by it. $h(h(t_1)$ means that it stores the data from the previous unit and multiplies it by that unit's mass. This means that t_1 must be some kind of storage system. After the values of the parameters are combined, sigmoid activation function is applied. In this case, sigmoid function would return numbers between 0 and 1.
Similar to the update gate formula, the reset gate formula utilizes a gate and weights that are different from the original.

$$r_t = \sigma(w)^{(r))}x_t + U^{(r)}h_t - 1\text{........}(ii)$$

Here, x_t and h_t serve as inputs, and the sigmoid function is applied after being multiplied by their respective weights and added together Using this method, we can determine the state of the memory gate. At the outset, a weight U is applied to the input x_t and a weight U is applied to the output h_t. Solve for the Hadamard product of the reset gate r t and Uh_t, element by element $(t-1)$. Here you may choose which time steps to deduct from the previous ones. The tanh nonlinear activation function is applied to the combined outcomes of steps 1 and 2. Finally, the network must compute h_t at the current time step in the final memory. The update gate is required in this situation. The current unit's data will be stored in and sent from the network through this vector value. The collected data will be selected based on what is now stored in memory $(h't)$ and what was collected in earlier timesteps (h') $(t-1)$. Component multiplication (Hadamard) is performed on the update gate and h(t-1) and then added to the Hadamard product of $(1-z_t)$ and $h'(t)$. The model's best guess for the update gate vector value is 1. In this instant in time, $1-z_t$ is quite close to zero, thus we may safely disregard the remaining feedback.

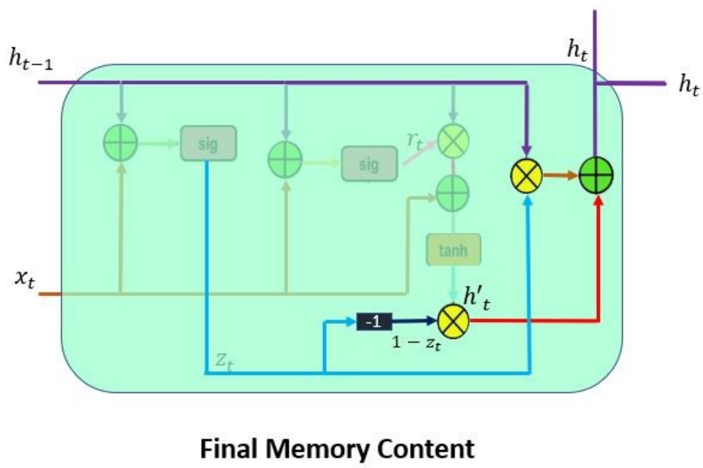

Figure 4.2: Final Memory Content [19]

Subsequently, z_t is accustomed to compute $1 - z_t$, which is then added to $h't$ to generate results.$h(t-1)$ and z_t are used to compute the Hadamard product. The output of the product is supplied into the pointwise inclusion with h't to generate the concealed state's end outcome.
The GRU and its gating method are briefly described in this guide. A model keeps proper information and communicates it to the next time step to avoid disappearing gradients. GRU excels in difficult tasks including speech recognition and synthesis, natural language processing, and deep learning.

The dimensions of the embedding vectors and the number of classes in the classification dataset are indicated by the hyperparameters embedding_dim and class_num, which were initially used to train our data. Then, a sequential model was initialized with a Keras sequential class. The model is then given the embedding layer, which is used to produce word embeddings to convert each input word into a dense vector representation. The output_dim parameter, which controls the dimensionality of the embedding vectors, is set to embedding_dim, while the input_dim parameter, which specifies the vocabulary size, is set to num_words. The embedding process is terminated by setting the trainable option to false. After the embedding layer and in between the GRU levels, dropout layers are added. This dropout is a regularization approach that, in order to avoid overfitting, randomly sets a function of input units to 0 during training.
Three GRU layers are added to the model, each of which is configured to return sequences and has a predetermined number of units (200, 300, 200). A GRU is a kind of recurrent neural network (RNN) that can record sequential data and is often used for problems involving sequence modeling. It is next essential to link the GRU layers to the dense layer, and they are added to the model. The first layer has 100 units and employs ReLU activation functions. A flattening layer is then added to the model, flattening the output of the previous layer into a vector. The ReLU activation function is also used in the second dense layer, which comprises 64 units. The final dense layer, which makes use of the softmax activation function and contains class _num units, generates the probability distribution over the classes. The model summary is printed by the model 3. summary() function, which also displays the kinds of layers, output shapes, and the overall number of trainable parameters. In order to produce predictions based on the learned representations, the model architecture seeks to capture the sequential patterns in the input text.

4.2 Few-Shot Learning

Few-shot learning is a specialized area within the broader field of machine learning and deep learning that seeks to impart the ability to AI models to learn from a limited quantity of labeled training data. Few-shot learning aims to facilitate the ability of models to extrapolate to unique, unobserved data instances by leveraging a limited set of samples provided during the training phase[13].

A significant obstacle in the field of machine learning pertains to the requirement for substantial quantities of training data that possess a balanced distribution of training samples in order to effectively train models. Machine learning models are capable of generalizing to new and unseen data samples based on their training with

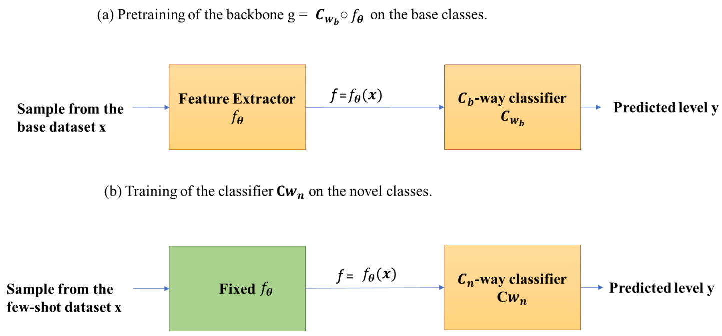

Figure 4.3: Illustration of Few Shot Classifier

large datasets. However, in practical scenarios, acquiring significant quantities of standardized data can prove challenging and require a considerable amount of time. Few-shot learning is a technique that is utilized in this context. This facilitates the utilization of machine learning algorithms with a limited amount of training data. Subsequently, the model acquires knowledge from a limited set of structured data and proceeds to label the remaining unstructured data samples.

Given that our dataset consists of textual data, we have selectively annotated 2,100 out of 10,000 data samples to facilitate the implementation of a few-shot learning algorithm. The purpose of this approach is to enable the algorithm to learn from the annotated data samples and subsequently annotate the remaining data samples. The manual leveling of 10,000 data samples is a laborious and challenging task. By employing few-shot learning techniques, it is possible to classify the remaining data samples[18].

4.2.1 Sentence Transformer (SBERT)

The Python library known as Sentence Transformer is capable of generating advanced embeddings for sentences, text, and images. These embeddings can be utilized for classification purposes employing cosine similarity between embeddings. This approach is particularly useful for identifying sentences with similar meanings, thereby facilitating semantic textual similarity, semantic search, and paraphrasing.

For example, we have employed four distinct divisions. Classes are: very negative, neutral, and positive, with 404,896,841 and 231 leveled data, respectively. Then, we developed a classifier using this minimal data. We created embeddings of our labeled dataset and, during inference, measured the distance between the new dataset and the embeddings of each category to determine its classification. During the process of generating embeddings for our dataset using SBERT, the dataset was segmented into multiple sentences. This is because SBERT operates at the sentence level, preserving contextual information and making comparisons accordingly. The SynTOK Library was employed to segment the document into numerous sentences and apply various regular expressions to partition the dataset into a sentence list.

Now that we have the list of phrases for the dataset, we will use SBERT to extract the embeddings. In this study, the all-minilm-l6-v2 model was employed in the algorithm, as this is faster and gives decent accuracy.

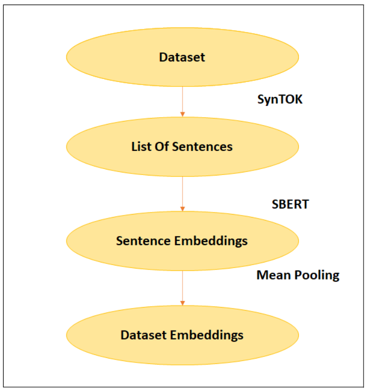

Figure 4.4: Flowchart of dataset embeddings

4.2.2 all-MiniLM-L6-v2

The all-MiniLM-L6-v2 sentence transformer is a neural network model that has been specifically developed to produce sentence embeddings of superior quality. Sentence embedding refers to a vector representation of a sentence that has a fixed length and is capable of capturing both its semantic and syntactic meaning. The mentioned sentence transformer is based on the MiniLM architecture, which is a variant of the SBERT model that uses fewer layers and parameters while maintaining high performance. During the training process, the all-MiniLM-L6v2 model is fed pairs of sentences and learns to make embedding vectors that accurately show how similar the sentences are. The model acquires this through training with contrastive learning, a self-supervised learning objective. Contrastive learning involves training the model to optimize the similarity between two augmentations of a given sentence. Once the model is trained, it can be used to generate sentence embeddings for new sentences, and these embeddings can be used for various tasks such as text classification. The quickness and precision of this sentence transformer render it a state-of-the-art tool for numerous natural language processing (NLP) applications. It is trained on a large corpus of text data using the same self-supervised learning that involves predicting missing words within sentences in order to learn general patterns of language.

We have used the all-MiniLM-L6-v2 sentence transformer as it is suitable for our dataset, and this model is better than other variants of SBERT in some ways,

- **Architecture:** The all-MiniLm-L6-V2 employs the MiniLm architecture, a model architecture that has been optimized for sentence embeddings and is known for its high efficiency and effectiveness
- **High Accuracy:** The all-MiniLm-L6-V2 sentence transformer is able to produce more accurate vector representations of sentences since it has been trained on a significant amount of text data
- **Versatility:** The model exhibits versatility in natural language processing (NLP) applications, as it can be fine-tuned for a range of tasks, including sentiment analysis, text classification, and question answering
- **Performance:** In comparison to all-mpnet-base-v2, the all-MiniLm-L6-v2 sentence transformer is five times faster and provides high quality

In general, the language model known as all-MiniLm-L6-v2 is a robust tool for developers seeking to construct sophisticated applications capable of comprehending and generating responses to natural language processing (NLP) inquiries with heightened precision and fluency

4.2.3 Data Augmentation

Data augmentation refers to the methods or procedures utilized to artificially expand the size of the training dataset by introducing modifications to numerous copies of the pre-existing dataset. This method is primarily undertaken to augment the dataset's volume or to integrate greater diversity within it. Insufficient data in the training dataset can have a significant impact on the accuracy of a machine-learning model. Collecting new data can be a time-consuming and costly process, further exacerbating the issue. Therefore, the implementation of data augmentation is necessary in order to mitigate the constraints posed by limited data. Enhancing the performance and outcomes of machine learning models can be facilitated through their utilization. The utilization of data augmentation techniques enhances the diversity and sufficiency of the data, resulting in improved and accurate performance of the model. The utilization of data augmentation techniques increases the likelihood of overfitting the model by producing supplementary training data and subjecting the model to diverse variations of data.

In our dataset, we have four classes of posts in total which are very negative, negative, neutral, and positive. We didn't have proper distribution of data for all of the four classes which was hampering our model's performance. We were facing a class imbalance issue due to limited data in a specific class. Previously, we had 4375 neutral posts, 4112 negative posts, 1155 very negative posts, and only 750 positive posts after labeling our data with few-shot learning. The number of positive posts was inconsistent with respect to other classes for which we were facing problems related to accuracy. After augmenting the dataset, the amount of positive data increased by 400 and became 1150. Hence, data in four of the classes were balanced.

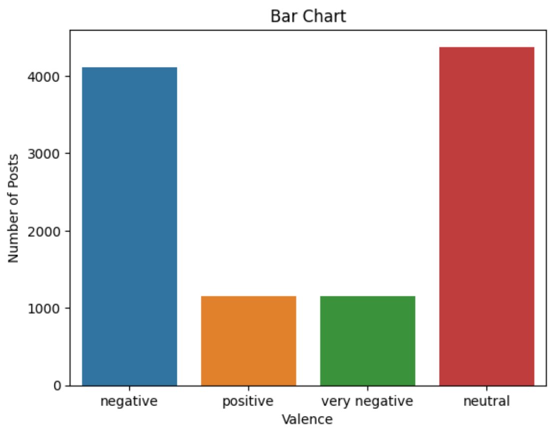

Figure 4.5: Data Distribution after augmentation

Chapter 5

Result Analysis

5.1 Precision, Recall, and F1 Score

Precision: Precision can be defined as the ratio of true positives to the total number of positive instances. The term ”positive predictive value” is also recognised in academic literature. Precision is employed in conjunction with recall to mitigate the effects of erroneous positive and negative classifications. The class distribution might have an effect on the precision. Precision will be reduced if there are a greater number of samples from the minority group. The level of exactness or quality that may be achieved is referred to as precision.

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} ....(i)$$

Recall: Recall is the measure of how well our model accurately finds true positives. The term ”true positive rate” is commonly used to refer to this metric. Also, it evaluates how well our machine learning model can differentiate between false and true positives inside a dataset.

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} ....(ii)$$

F1 Score: F1 score is a measure that can combine both precision and recall. Due to the connected trade-off between precision and recall, F1 may be applied to evaluate the efficacy of our models. A small value of precision or recall would lead to a reduced aggregate score. Therefore, it aids in achieving equilibrium between the two metrics.

$$F1 = \frac{2(Precision * Recall)}{Precision + Recall} ....(iii)$$

5.2 Result

Our results were evaluated using four distinct methods. The subjects were categorized into four different categories, namely "neutral," "positive," "very positive," and "negative." The determination of the outcome is achieved by employing diverse metrics, including accuracy, precision, memory, F1 score, and multiclass classification, which are utilized in the analysis of the confusion matrix. The subsequent outcomes are derived from the utilization of GRU and Few Shot techniques.

Classifications	**Precision**	**Recall**	**F1 Score**
Neutral	0.77	0.80	0.79
Positive	0.50	0.62	0.56
Negative	0.71	0.68	0.70
Very Negative	0.67	0.60	0.63
Macro average	0.67	0.67	0.67
Weighted average	0.71	0.71	0.71

Table 5.1: Few-Shot Result

The total accuracy of the model in the above scenario is 0.71.The macro average is a metric that denotes the mean level of performance across all classes. Overall, the macro average of the F1 score is 0.67. Instead, a weighted average is calculated by adding all the account support from each class. The class's evaluations are affected by this. The weighted average value of the F1 score is 0.71.

Classifications	**Precision**	**Recall**	**F1 Score**
Neutral	0.81	0.83	0.82
Positive	0.72	0.39	0.51
Negative	0.71	0.83	0.76
Very Negative	0.75	0.73	0.74
Macro average	0.75	0.70	0.71
Weighted average	0.75	0.75	0.74

Table 5.2: GRU Result

Here, the overall accuracy of the GRU model is 0.75. The F1 score of the macro average value is 0.71. It is the outcome of the model's average performance of all classes. The weighted average of the F1 score value is 0.74 which takes the support of each class.

5.3 Confusion Matrix

In the context of evaluating the effectiveness of a classification model, a confusion matrix is employed. This matrix is of dimensions N-by-N, where N denotes the overall count of target classes. The matrix facilitates a juxtaposition between the pragmatic objective values and the prognosticate values generated by the machine learning algorithm. A very common way to solve classification problems is to use a confusion matrix. This methodology is applicable for classification tasks involving both binary and multi-class scenarios. The determination of true values for test data is a prerequisite for accurate determination.

<table>
<tr><td rowspan="2" colspan="2"></td><td colspan="2">Predicted</td></tr>
<tr><td>Negative (N) -</td><td>Positive (P) +</td></tr>
<tr><td rowspan="2">Actual</td><td>Negative
-</td><td>True Negative (TN)</td><td>False Positive (FP)
Type Error |</td></tr>
<tr><td>Positive
+</td><td>False Negative (FN)
Type Error ||</td><td>True Positive (TP)</td></tr>
</table>

Figure 5.1: Confusion Matrix[11]

The confusion matrix is a tabular representation that displays the number of correct and incorrect classifications made by a classifier.. It is used to see how well a classification model works. A confusion matrix can be used to determine how well a classification model figures out the calculation of accuracy, precision, recall, and F1 score. Confusion matrices are mostly used because they can give a model better success. For instance, assume you have data that has only two classes and 89% of the data is in Class X and 11% of the data is in Class Y. Also, imagine your classification model identifies properly all the instances of Class X and incorrectly identifies all instances of Class B. Here, the accuracy of the model is 85%. On the other hand, class B is incorrectly classified which is not acceptable.[14].

True Positive: The true positive metric evaluates the effectiveness of the model in predicting instances belonging to the positive class. The model suggests that the given instance will yield a positive outcome, and subsequently, this outcome is observed. Accurately identifying positive cases is a crucial aspect when evaluating the predictive performance of a model. In practical scenarios, a true positive is identified when the model accurately predicts the classification of an email as spam. The true positive rate refers to the frequency with which an instance is accurately classified into a specific category. Based on the confusion matrix provided, it can be observed that among the total of 98 actual positive cases, 95 were accurately predicted as positive, resulting in a True Positive value of 95.

31	3
3	95

Table 5.3: Example of Confusion Matrix

False Positive: When a model incorrectly identifies an instance as belonging to a certain class, this is known as a false positive. It is also known as a Type 1 error. False positives occur when our model incorrectly predicts a negative result. You assumed a negative number, but it turned out to be positive. For example, a woman tested pregnancy positive but actually she is not. The overall accuracy of a classification model might be impacted negatively by the presence of false positives.

False positive refers to the quantity of negative instances, out of a total of 34, that were erroneously classified as positive. Three of the 64 true negative instances were erroneously classified as positive. Thus, the value of False Positive is 3.

True Negative: True negatives refer to the outcomes that the model accurately predicts as negative. How many times have we counted actual negative values and those values are predicted as negative values? Whenever you predict a negative value, it is correctly a negative value. A true negative value is correct when there is a system or tool that has no threat or vulnerability. For example, if an antivirus software identifies that a file is virus free and it is totally clean it can be considered as a True negative. A classification model's accuracy may be evaluated in part by counting the number of false negatives it produces. In most cases, good model performance is indicated by a high number of true negatives. There were 34 true negatives, and 31 of them were accurately predicted. In this case, the true negative has the value of (31).

False Negative: In the context of predictive modelling, the term "false negative" refers to the situation where the model predicts a negative outcome, while the actual outcome is positive. It is also known as Type 2 error. For example: When an antivirus program fails to detect a virus in a file, although the file is infected. Computer systems and networks are vulnerable to damage from such results. The consequences of false negatives are generally more severe than those of false positives, thus they must be taken into consideration when evaluating the effectiveness of a classification model. Only 3 true positives out of 98 are wrongly identified as negative. Therefore, a false Negative has a value of (3)[21].

5.3.1 Confusion Matrix-Based Result

We can derive the levels of arousal—neutral, positive, negative, and very negative—from the comments on the confusion matrix. Our confusion matrix demonstrates the differences between the predicted and actual values. The types of content in both forms are the same. The primary comparisons made by the confusion matrix are those between the true positive, true negative, false positive, and false negative. Our true values come into four categories: True negative, True very negative, True neutral, and True positive. Here is one of our confusion matrix tables attached below:
From the matrix table, the true values are 67 (True Negative), 69 (True Neutral), 13 (True Positive), and 30 (True Very Negative). All the values indicate the actual values and other values of the table are false values in our case.
From Figure 5.3, true values are 339 (True Neutral), 58 (True Positive), 78 (True Very Negative), and 293 (True Negative).

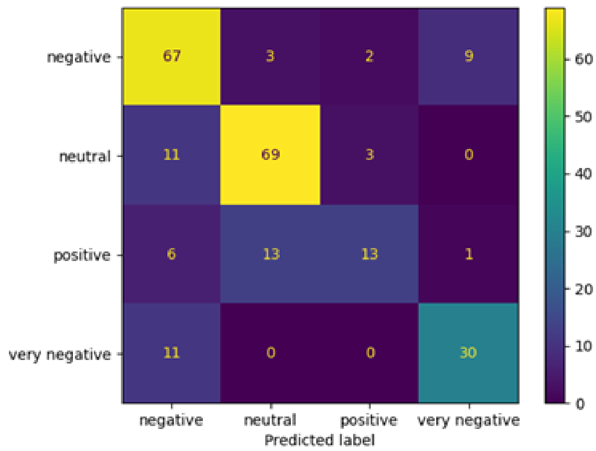

Figure 5.2: Few shot Confusion Matrix

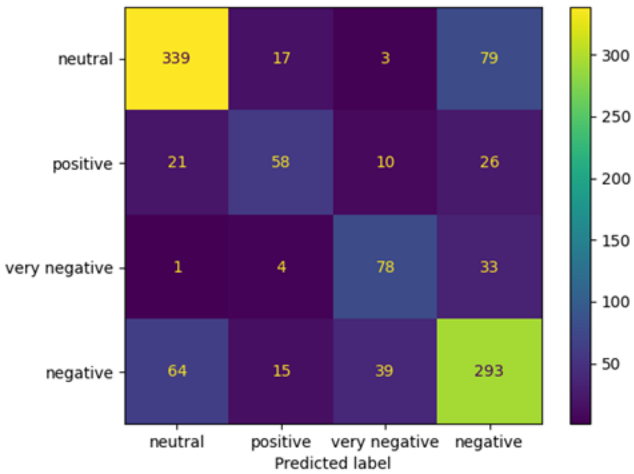

Figure 5.3: GRU Confusion matrix

5.4 Result Comparison

Similar research by Mandar Deshpande and Vignesh Rao (2017) entitled "Depression Detection Using Emotion Artificial Intelligence" In their paper, they detect depression using the Multinomial Naive Bayes and Support vector machine models. They obtained Multinomial Naive Bayes scores of 0.83 for recall, 0.83 for precision, 0.8329 for F1 score, and 0.83 for overall accuracy. The SVM Recall score was (0.79), the precision score was (0.804), the F1 score was (0.7973), and the model's accuracy is 0.79. For the few-shot model, we obtained recall (0.83), precision (0.81), F1 score (0.82), and accuracy (0.75). The comparison of the top results from both studies is provided below:

Models used	Data Set Type	Data Size	Best Recall	Best Precision	Best F1 Score	Best Accuracy
2(Mandar D., Vignesh R.)	Binary	10000	0.83	0.836	0.8329	0.83
2 (Our)	Multiclass	10087	0.83	0.81	0.82	0.75

Figure 5.4: Comparison Table

5.5 Future Proposed Work

In our forthcoming work, we aim to persist in our data collection procedure and annotation to augment the quantity of data for training objectives in our dataset. Additionally, we aspire to produce high-quality data by implementing appropriate pre-processing techniques and endeavor to balance our four data classes suitably through data augmentation. Enhancing accuracy would be facilitated. Furthermore, we seek to seek the advice of experts regarding our data annotation. Given that it pertains to mental health, it is advisable to seek assistance from professionals in the field of mental health. Subsequently, the reliability of our annotations will significantly increase. Furthermore, there is a desire to augment the variety of cancer types for which patients can achieve survival, with a minimum target of five distinct types. Additionally, we aim to gather data through direct communication methods such as face-to-face interactions and virtual meetings. We aim to conduct a thorough examination of the emotions and thoughts experienced by individuals during that period, with the goal of enhancing our comprehension of their emotional states beyond what has been previously documented. Consequently, this will enable us to produce high-quality data annotations. In addition, this study made use of two machine learning models, namely few-shot learning and Gru, for the analysis of the datasets. In our forthcoming research, we aim to leverage hybrid models through the amalgamation of various models used during our training phase, with the objective of creating a unified machine-learning model.

Chapter 6

Conclusion

Throughout this research, we have shown that a variety of social media platforms are also utilized to evaluate and identify significant depression among cancer patients with the lowest survival rates. In order to provide a full picture of our work, several research topics were covered. Using a dataset for mental health identification, the accuracy of several classifiers in identifying mental health is compared. The analysis done on the chosen dataset sheds light on the research concerns. Here is a rundown of the things we have learned: The definition of mental health detection as well as its frequent causes, including depression, anxiety, etc. Although gloomy, melancholy, or negative emotions are prevalent, few people experience them regularly, over extended periods of duration(weeks, months, or even years), and often without obvious explanation. Despondency may be a true illness that affects a patient's physical and mental emotions, as opposed to being only a state of mind. Depression may affect any individual at any moment. However, our circumstances might occasionally heighten our vulnerability to despondency. Getting old, the death of a close friend, establishing a family, and retiring all result in somatic and emotional alternations that, for a tiny number of individuals, may be depressing. We studied the unrestrained process, the secular process, and the linguistic style, and we oriented a model to operate each kind of component alone and in conjunction with the others. We classify the qualities of texts using approaches from machine learning. Our research indicates that, on average, each classifier achieves accuracy in the 60-75% range. We want to employ a more cutting-edge method to extract phrases from a wider range of emotional traits in future research. We would also want to verify the usefulness and effectiveness of our strategies using other datasets. We concur with the body of existing data that targeted research on mental health analysis is required. The emotional process element was shown to be the most essential characteristic.

Bibliography

[1] M. Deshpande and V. Rao, "Depression detection using emotion artificial intelligence," in *2017 international conference on intelligent sustainable systems (iciss)*, IEEE, 2017, pp. 858–862.

[2] A. U. Hassan, J. Hussain, M. Hussain, M. Sadiq, and S. Lee, "Sentiment analysis of social networking sites (sns) data using machine learning approach for the measurement of depression," in *2017 international conference on information and communication technology convergence (ICTC)*, IEEE, 2017, pp. 138–140.

[3] M. R. Islam, M. A. Kabir, A. Ahmed, A. R. M. Kamal, H. Wang, and A. Ulhaq, "Depression detection from social network data using machine learning techniques," *Health information science and systems*, vol. 6, pp. 1–12, 2018.

[4] M. Ketcham, M. Piyaneeranart, and T. Ganokratanaa, "Emotional detection of patients major depressive disorder in medical diagnosis," in *2018 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, IEEE, 2018, pp. 332–338.

[5] J. K. Rout, K.-K. R. Choo, A. K. Dash, S. Bakshi, S. K. Jena, and K. L. Williams, "A model for sentiment and emotion analysis of unstructured social media text," *Electronic Commerce Research*, vol. 18, pp. 181–199, 2018.

[6] K. Sailunaz, M. Dhaliwal, J. Rokne, and R. Alhajj, "Emotion detection from text and speech: A survey," *Social Network Analysis and Mining*, vol. 8, pp. 1–26, 2018.

[7] P. Soleimaninejadian, M. Zhang, Y. Liu, and S. Ma, "Mood detection and prediction based on user daily activities," in *2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)*, IEEE, 2018, pp. 1–6.

[8] M. M. Tadesse, H. Lin, B. Xu, and L. Yang, "Detection of depression-related posts in reddit social media forum," *IEEE Access*, vol. 7, pp. 44 883–44 893, 2019.

[9] H. Nunoo-Mensah, *Graph showing the disparity of research in emotion detection and... — download scientific diagram*, https://www.researchgate.net/figure/Graph-showing-the-disparity-of-research-in-emotion-detection-and-emotion-detection-from_fig1_341491298, 2020.

[10] C. S. A. Razak, M. A. Zulkarnain, S. H. A. Hamid, N. B. Anuar, M. Z. Jali, and H. Meon, "Tweep: A system development to detect depression in twitter posts," in *Computational Science and Technology: 6th ICCST 2019, Kota Kinabalu, Malaysia, 29-30 August 2019*, Springer, 2020, pp. 543–552.

[11] A. Suresh, *What is a confusion matrix?. everything you should know about... — by anuganti suresh — analytics vidhya — medium*, https://medium.com/analytics-vidhya/what-is-a-confusion-matrix-d1c0f8feda5, 2020.

[12] D. J. S. R. Dr. S. Smys, *03.pdf*, https://irojournals.com/tcsst/V3/I1/03.pdf, 2021.

[13] D. Saini, *Few shot learning using sbert. document classification with sbert — by deepak saini — analytics vidhya — medium*, https://medium.com/analytics-vidhya/few-shot-learning-using-sbert-95f8b08248bf, 2021.

[14] Simplilearn, *What is a confusion matrix in machine learning?* Feb. 2023. [Online]. Available: https://www.simplilearn.com/tutorials/machine-learning-tutorial/confusion-matrix-machine-learning?fbclid=IwAR0ICS6q78TFh_vX1SPoz1o1WMJNpjyj9YbvoJgN0vY5XwG1RM8dcUMtIBk.

[15] *10.2. gated recurrent units (gru) — dive into deep learning 1.0.0-beta0 documentation*, https://d2l.ai/chapter_recurrent-modern/gru.html?fbclid=IwAR25kybNRZ_gyMQykJzNv18IhG6hMDId2HhqdPoG_3YR4qTDCB1B8Bc4ChY.

[16] *A average recognition rate of transitional emotion type for... — download scientific diagram*, https://www.researchgate.net/figure/a-Average-recognition-rate-of-transitional-emotion-type-for-subject-independent-result_fig8_315878863.

[17] *Cancer survival rates — nuffield trust*, https://www.nuffieldtrust.org.uk/resource/cancer-survival-rates, (Accessed on 05/29/2023).

[18] *Few shot learning using sbert. document classification with sbert — by deepak saini — analytics vidhya — medium*, https://medium.com/analytics-vidhya/few-shot-learning-using-sbert-95f8b08248bf.

[19] *LSTM versus GRU Units in RNN — Pluralsight — pluralsight.com*, https://www.pluralsight.com/guides/lstm-versus-gru-units-in-rnn?fbclid=IwAR13kixN5xPp39mEwjHUKnZ5rnBYhBl2GZ4Gsn1l3fqroVY9OsPKa7mYWTU.

[20] *Text based emotion recognition techniques. — download scientific diagram*, https://www.researchgate.net/figure/Text-based-emotion-recognition-techniques_fig1_346265290.

[21] *What is a confusion matrix in machine learning?* https://www.simplilearn.com/tutorials/machine-learning-tutorial/confusion-matrix-machine-learning?fbclid=IwAR0ICS6q78TFh_vX1SPoz1o1WMJNpjyj9YbvoJgN0vY5XwG1RM8dcUMtIBk.