USING DATA ANALYSIS AND MACHINE LEARNING FOR STUDYING AND PREDICTING DEPRESSION IN USERS ON SOCIAL MEDIA

by

CHANPREET SINGH

A thesis submitted to the Faculty of Graduate and Postdoctoral Affairs in partial fulfillment of the requirements for the degree of

Master of Information Technology

in

Digital Media with a Specialization in Data Science

Carleton University
Ottawa, Ontario

Abstract

Mental health problems leading to depression have become a critical concern due to the towering engagement of people on social media platforms. Several past approaches have been implemented by analyzing the pattern, behaviour, and vocabulary of the posts by users on social networking sites. This research proposed a system to predict users who could have been affected by depression, by introspecting characteristics of users already being affected. A combination of both the tweet-level and the user-level architecture was used to generate a more robust and reliable system where semantic embeddings trained from advanced neural networks were adopted under the tweet-level, whilst for the user-level, an approach using 12 significant features was operated by extensive feature engineering. Further, SVM with Word2Vec and TF-IDF under tweet-level yielded an accuracy of 98.14% and recall of 95.63%, whereas the gradient boosting classifier under user-level revealed an accuracy of 95.26% with a recall of 86.75%.

To my parents, for their love and support.

Acknowledgements

First and foremost, I would like to acknowledge my indebtedness and render my sincerest gratitude to my supervisor, Prof. Dr. Omair Shafiq, for the enthusiastic encouragement, useful critiques, patient guidance, eminently capable supervision, and immense knowledge he has provided throughout my two years of master's degree. Dr. Omair Shafiq has always been a keyperson for me to inculcate the learning attitude towards cutting-edge research since the commencement of my graduate studies. His extensive, encouraging and productive feedbacks, positive confidence in my research capabilities encouraged and inspired me to explore beyond the limits, thinking out of the box, and organize my time wisely. His detailed feedbacks and careful editing contributed enormously to my research and eventually this thesis. I am extremely thankful for our friendly and thoughtful discussions at the end of our research meetings and for his support in my academic, as well as career guidance. It is whole-heartedly appreciated that his great supervision proved monumental towards the success of my research and overall studies.

I am highly obliged in taking the opportunity to sincerely thanks Dr. Ali Arya, Dr. Audrey Girouard, Dr. Olga Baysal and Dr. Frank Dehne for helping, as well as supporting me directly or indirectly throughout my journey in graduate school. I am thankful for your leadership, encouragement, creative and comprehensive advice during my academic years, which contribute to complete this endeavour and made this research work possible.

I would also like to extend my deepest gratitude to the entire faculty of Graduate and Postdoctoral Affairs (FGPA), and staff members of faculty of the School of Information Technology for their generous attitude, friendly behaviour, helping with various resources, as well as opening windows of opportunity for the future to us.

Table of Contents

List of Tables

List of Illustrations

LIST OF ABBREVIATIONS

CES-D	Center for Epidemiologic Studies Depression
CSV	Comma Separated Value
ISADS	International Symposium on Autonomous Decentralized System
LDA	Latent Dirichlet Allocation
LSA	Latent Semantic Analysis
LSTM	Long Short-Term Memory
MDD	Major Depressive Disorder
NLP	Natural Language Processing
PCA	Principal Component Analysis
PHQ-9	Patient Health Questionnaire 9
PTSD	Post-Traumatic Stress Disorder
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic
SMDI	Social Media Depression Index
TPR	True Positive Rate
TNR	True Negative Rate
PPV	Positive Predictive Value
NPV	Negative Predictive Value
FPR	False Positive Rate
FDR	False Discovery Rate
FNR	False Negative Rate

Chapter 1: Introduction and Background

This chapter explains the influence of social media in daily routine of users as well as ramifications of how social media contributes to depression, along with the description of depressive behaviour symptoms, methods to detect as well as diagnose depression. Further, utility of data analytic tools to develop and predict statistics from social media posts are discussed. Lastly, a summary of this dissertation outlining details from all chapters is portrayed.

1.1 Introduction

We are entering an era of future technology where digitalization has been burgeoning on a hefty scale. To add to it, Internet has connected the world more important than ever by spawning the person's individuality into a cybernetic space where they could socialize with a large number of people on the same platform. Furthermore, a large number of people spend most of their time using social media as a daily routine to post their latest updates, feedback as well as opinions and comments publicly by using platforms such as Twitter, Facebook, Tumblr, Snapchat, YouTube, Instagram, etc. generating an enormous amount of information. However, this information has been proved vital source for a lot of marketing organizations, big business industries, telecommunication, and other big data running companies by hiring data scientists to analyze a lot of hidden facts, as well as statistics by preprocessing that information into a meaningful database. Further Compromising that the internet has been taking control over the mindset of people to such an extent that it has resulted in causing a lot of mental health problems resulting in depression. However, these problems dealing with context to occurrence of factors such as

anxiety to stay update every time, less privacy in life, cyberbullying, fear of missing out (FOMO), social media jealously along with many other reasons that has been targeted by various health organizations as well as researchers to investigate the role of social media by manipulating the correlation between depression in users, and its causes that revolves within the information available publicly. Moving further, an article issued by the World Health Organization on 30th January 2020 provides awareness on depression and shows statistics of over 264 million people that are undergoing depression belonging to all age groups more of which are female in count [58].

Depression has been proved to be one of the biggest causes of people committing suicides recorded to be 800,000 people every year [58]. However, most of the cases belong to a range between teenagers of 15 years to 29 years of adult [58]. What is more, there exist a situation among the people suffering from any mental illness where lack of awareness for a professional person to consult with and type of treatment to get, manipulates the count of pending cases for which depression gets detected but not given a required therapy. Some of the reasons behind this condition can be due to lack of proper communication of the patient to convey their emotions, the consulted person or doctor is not a certified mental health professional such as a psychiatrist or a psychologist and other limitations where a lot of cases remain obscure due to the fear of being judged or expenses for getting a proper medication. Further, these conditions arise two cases under which for the first case, if the depression was detected, but not given a proper medication and second case is due to the lack of awareness of the type of person to consult for such medical illness, both cases will ultimately have a negative impact on the health of the patient. To instantiate, if a person is showing symptoms of depression there could be many complications to understand the

reasons behind not getting appropriate countermeasure that includes factors such as false diagnosis of symptoms resulting in false detection for type of depression in patient leading to an improper or wrong medication, inadequate expenses to invest for a psychological Treatment as well as circumstances where even if the depression gets diagnosed but due to the lack of proper medication facilities remains untreated.

1.2 Depression Diagnostic Criteria

The Diagnostic and Statistical Manual of Mental Disorders 5th edition (DSM-5) [107] mentions about the diagnostic criteria for major depressive disorder under which it is stated that if a person has five or more of these symptoms mentioned below for a consecutive period of two weeks and there is a noticeable shift from previous functioning levels, then the person is suffering from major depressive disorder.

The symptoms for diagnostic criteria of major depressive disorder are as following:

1. Depressed mood for a whole day [107].
2. Gains no pleasure in performing the usual activities and at the same time, there is lack of interest in doing the everyday tasks.
3. There is a remarkable increase or decrease in the weight and appetite [107].
4. Nearly each day person suffers from insomnia or hypersomnia [107].
5. psychomotor agitation or slowing down of movements [107].
6. Feeling tired or lack of energy daily.
7. Experiencing guilt and feeling worthless every day.
8. Lack of ability to concentrate.
9. Suicidal thoughts or attempt to suicide

1.3 Social media and Depression

Nevertheless, with the evolution of new medical advanced methods and technologies people with an untreated depression or people who are hiding their mental illness due to so many reasons can be taken over on a machine learning architecture with the use of social media as a single platform that frames everything on a publicly exposed trajectory. To elucidate, billions of people are connected worldwide with the use of internet where countless conversations, data uploads and downloads are taking place every second, this content and information is responsible for a lot of data analysis that helps to generate useful information for various sectors such as tele-communication, business, health, government security as well as other multinational and technical sectors.

Further, there are many types of internet platforms to surf, post, explore, catch various new features and techniques which influence the pattern of a user responding to a particular web space such as Social Media Websites, YouTube, Blogging, Online Shopping along with other online developer environment tools etc. Each of these platforms portray a different type of user analytics that comes under pragmatic application for various other technological industries.

To elucidate, if we select an advertisement and buys the product shown on Facebook ads, ultimately the meta tags and other constraints will recommend the next advertisement to be similar from the previous one, this is because of a pattern by the user, gets observed and analyzed by the machine learning algorithms working at the background system. Also for an online-shopping website, if the user mostly browse in the account at night hours and buys more products at night, then the advertisements also learn a pattern of user time engagement to show more of the new items similar to the previous purchased

products only during the night time because of various factors which include hover time on the items browsed, clicked items, selected items, wish list in addition to previous purchased items category.

In the same direction, by utilizing the power of data analytics, a technique can be developed that helps to predict various insights among users on social media. This technique works on a configuration that if we break down the structure of users according to their posts, comments, tweets, type of place of work, location, eating habits, tracks of locations travelled along with other publicly posted data which can be observed mostly under the social media websites such as Instagram, snapchat, Facebook, twitter and Tumblr, Various future unfavorable as well as beneficial outcomes can be detected. To exemplify, a data collection by user posts using social media with locations travelled, food and hotel selection, along with other related information can be analyzed by tourism and travel industry to get various statistics which can help to upgrade the costs, food, customer needs and many other aspects. Similarly, a pattern of user behavior by building a sentiment analysis model can be useful to predict various future outcomes such as positive, negative and neutral interest regarding any trending topics.

On top of that, this research emphasizes on building a novel architecture for predicting depression, a major health concern rising under users on social media. Further, the initial procedure requires to select an appropriate data source to target the amount of real time user information, which is publicly available. However, amongst all the social media platforms, Twitter proves to be the best source for data collection due to its open source attributes as well as higher compatibility of aspects such as time, speed and connectivity on a developer level API processing. Moreover, Twitter user posts such as

comments and retweets use a 280-character approach which makes them easier to analyze as compared to other platforms that provides a larger input limit. Also, contrasting with the other social media platforms, users on twitter use less direct messaging service which makes mining of tweets easier due to retweets and comments, which helps to obtain a dataset that contains more sensitive and personal information to be analyzed for interpreting the user behaviour.

To encapsulate the overview of this research, the introduction and background study helps us understand why depression is a major problem, the factors causing depression, the methods on which we can detect depressive users utilizing social media and lastly the importance of twitter as a preeminent source of information. After selection of a data source, the literature review and comparative analysis focus on an in-depth-exploration of previous research papers, reviewed to highlight the key aspects of past work done in formation of a comparative table. The comparative analysis will help to understand the key features used in previous research such as methodology, data collection, strength, weakness, research motive and results.

Also, this part of the paper will help us to finalize the concepts, enhance the creativity to think differently and come up with a novel idea of developing a unique model. Further, the dataset collection is unique in its own manner due to a two-way approach that will help to illustrate the statistics behind a tweet to tweet as well as a user to user level architecture.

Nevertheless after collecting the data, a gap analysis with an opinion towards the problem statement and questions related to the purpose of this research are formulated, which overall helps to develop an idea of unique factors that will ultimately contribute towards new statistics, extensive feature engineering, change in existing algorithmic

patterns, integrated together for further experiments as well as proposed solution. Moving next to the experiments, where tweet to tweet level architecture was analyzed by performing various approaches to improve the outcome. Followed by the experiments, the next stage is the proposed solution for which 12 novel key features were designed in such a manner that the model outperform the results from previous researches which was verified in the evaluation stage with various metrics such as accuracy, time complexity, recall, precision and F1-score along with an implementation of k-fold cross validation for countenance of the outcomes. Last but not the least, the outcomes help to solve the research questions discussed in the conclusions from this research. After the conclusions, the ending of this research will delineate some of the future work techniques and concepts that will be developed as per according to the different datasets as well as combining of a more complex hybrid architecture or an AI-operated multimodal feature engineering structure.

Chapter 2: Literature Review and Comparative Analysis

This Chapter will discuss about background study of most relevant research works related to this research in context to their objective, methodology, results, strengths and weakness. Further, cover the details related to improvements that could be carried in previous studies for future application. Lastly, formation of a comparative analysis based on the literature review in a tabular form was carried out for understanding a clearer picture of fundamental factors in past research papers.

2.1 Literature Review

To begin with, one of the most important research papers research by Munmum De Choudhury et al [1] Aiming at Predicting Depression via Social Media [1] describes a severe issue that has been affecting the lives of millions of people in their personal as well as professional environments called “depression”. In their research, they study social media to investigate the detailed architecture for which they analyze the total number of people going through depression as compared to the ratio of people not able to get proper treatment by using crowdsourcing [1]. The conducting of surveys as well as collecting twitter profile access from users which were detected to have some sort of symptoms of depression helped by their surveys getting into a match and then extracting tweets from those profiles which were publicly open. With the increased use of the internet, people are sharing their personal and professional experiences on social media. The authors want to explore the potential of social media to predict the problem of depression by analyzing the activities of people who are already affected by it. They found that with the increased effect on mental health,

people begin to sideline themselves from social activities, have negative thoughts, have medicinal concerns, etc.

They have measured depression in the form of the disorder, which is MDD, Major Depressive Disorder [1]. They crowdsourced workers to conduct CES-D (Center for Epidemiologic Studies Depression Scale) screening test [1] from the Twitter users who were affected by MDD. They removed noise from the dataset, which means they removed the records which they believe could give bias results. The authors also collected history about the patients. They extracted various features out of the information available like volume, replies, questions, activation, followers, dominance, etc. [1]. They used a supervised learning approach, and out of tried supervised algorithms, the support vector machine algorithm gave the best result for their dataset. They trained and tested their models by combining all the features, individual features along with reducing the dimensionality of features also. They further verified their results with the t-statistical test.

The strengths of [1] include a detailed analysis of the real-time factors that can impact the mental health of a human being. The use of a large number of features and investigating the impact of each feature is the main key point of this paper. To make it more practical as well as generic in terms of finding a useful result they used real-time data collected over a series of times because the more the dataset applies to model results in better accuracy as well as better ideas to be implemented as future work. The weakness for [1] could be the authors not trying to measure depression in any other form other than MDD which locks down the other factors to get compromised such as anything related with a small stressful situation, a temporary depressed mood, anxiety, or having a bad day feeling gets posted in the form of textual or any other emoticons used in a post gets unmonitored.

However, the number of user samples which is 476 is not enough to generalize the depression behavior across all populations. Also, large and diverse samples across diverse age groups can help to study different behavioral traits among different age groups and gender. Most twitter users use emoticons to express their feelings, which helps to identify the present state of the emotions of that user and can also be included as one of the features.

The authors Munmun De Choudhary [2] and colleagues describe depression as one of the fastest-growing mental health problems. The paper aims to investigate the levels of depression present among individuals using their social media posts. Their methodology included collecting user's data from twitter and applying supervised learning models and then used social media indexing [2] to rank their depression levels. They collected data by building AMT platforms and using crowdsourcing technologies, they populated their data files. They used the CES-D (Center for Epidemiologic Studies Depression Scale) questionnaire [2] to rate the intensity of depression. They build up a positive class, containing data of depression affected users and a negative class, containing data of users that have the negligent effect of depression. Several features were also derived to categorize posts. These features were based on two approaches: post-centric and user centric.

Post features utilize several characteristics of the posts like time at which the tweet was posted, the language used in tweets, and the emotion described the user using emojis or other similar words [2]. User-centric posts include the interaction of a user with other users and the network generated by the user. For training their model, they used a supervised learning approach. To deal with the problem of a large number of features, they used PCA. After dealing with dimensionality, they applied the Support Vector Machine

Algorithm with RBF kernel and five-fold cross validation [2]. They trained their model using different features, combining all features, as well as by reducing the dimensionality. The important results that can be inferred from this research are that the models using linguistic features alone have better precision and recall rates [2]. The next finding found that the emotions change during the entire course of the day. The users with the positive class are more active during the nighttime. For determining the SMDI (Social Media Depression Index) [2], they used numerical quantities such as standard deviation and mean of depression-related posts, shared within the specific timeline and calculated the numeric value of SMDI. The range of the index is directly proportional to the severity by which the person is affected by depression. To instantiate, more index represents a higher magnitude of depression. Strength for [2] is when after determining the degrees of depression, they performed analysis using various factors. They downloaded twitter posts daily, for specific cities for a year, and tested the depression index present in those cities. They compared their results with the already existing list of non-happy cities in the USA and obtained a linear graph [2] for that comparison which validates their research conducted.

For the weakness in [2], there are several findings indicated by this paper, however, the limitations include that there are many features that can be inferred from this study, by studying patterns of the user activity, the evolution pattern of depression posts and the language used by affected patients in their preliminary stages. Also, as mentioned by the authors [2], few people on the internet use private twitter, therefore it is difficult to track the activities of all the users and diagnose this disease. In addition to Twitter, there can be an incorporation of more social media platforms like Facebook, Instagram in which people tend to share their views.

Further, in a research paper [3] the authors propounded the concept of jealousy and covetousness that has been introduced by the deliberate usage of social media platforms. Most of the social media users incline to share their personal and professional updates in a way that makes them look ahead of other people. This scenario created a hypothetical race among individuals and created a sense of competition among users. This constant push to look best in front of others, sometimes have negative impacts on mental health including depression. The authors discussed diary and experimental studies in which different users were taken into account, and interrelationship of Facebook use with depression was formulated [3]. The diary studies include the study conducted by Steers and Colleagues [3][11] in which they came up with the result that cynical attitude in the social media comparison can lead to depression among individuals.

One more study they discussed was by Verduyn and colleagues [3][12] in which users were asked to report their personality traits like well-being and jealousy, many times a day, for almost a week. They also found similar results that the well-being of a person changes if it is constantly influenced by the social media posts that constantly undermines them. As per [3], the experimental studies were also brought up to see a similar pattern. The study also highlighted the comparison related to looks and attractiveness. The women were found, comparing their beauty and looks with other women by analyzing social media posts of their peers and have a feeling of less contention if there is something inferior in them. The study conducted by Vogel and colleagues [3][13], also suggested that there is a decline observed in self-confidence and self-esteem in individuals if they browse feeds of other individuals who find more successful.

One more study indicating an effect on self-esteem by Vogel and colleagues [3][14], found that the social competition arousing among individuals caused a bad effect on a healthy state of mind. This type of social media environment not only brings instability to human health but also creates a sense of desire to lead the high-profile lifestyle as lead by their peers. This fact was illustrated by the study done by Appel and colleagues [3][15]. The pattern of usage of social media was also the reason behind the jealousy and discontentment among people. Verduyn and colleagues [3][12], found that people who use Facebook less actively, which means they only see other's posts and themselves are less involved in posting their updates have a sense of inferiority complex.

Strengths for [3] is the paper including all aspects of mental health and the emotions associated with it like envy, social comparison, inferiority complex, etc., which is the hidden outcome of depression and difficult to estimate by consuming social media data. The paper also leaves open-ended questions for future research and suggests future research to be conducted, that includes various social media platforms, incorporating several forms of depression, and consideration of other web blogs and user-centric platforms to be used for determining the levels of depression.

On the contrary, the paper [3] lacks the graphical explanation that could help to visualize the results more precisely. The paper should incorporate tables, graphs or diagrammatic information that helps to understand the difference of all the factors discussed in different researches.

Apart from this, the authors in the research paper [4] commenced with the problem of cases of depression that have remained undiagnosed due to various reasons and factors. The authors reviewed different papers and research pathways that deal with the detection

of depression using social media platforms like Facebook, Twitter and elucidated their methodologies along with results. This paper [4] includes the theoretical study of all the approaches used in various research. The research includes the paper of Reece et al.[4][24] who predicted user depression and post-traumatic stress disorder (PTSD) status from text and Twitter meta-data that with relatively high Areas under the Receiver Operating Characteristic (ROC) curve (AUCs) of 0.87 (depression) and 0.89 (PTSD). Tsugawa and other authors [4][10] predicted depression from streaming data from twitter collected from Japanese correspondents, using several assessment criteria. In [4][25], Reddit posts were utilized to study the mental well-being of university students. A prediction model was implemented on data gathered from Reddit mental health support communities and applied to the posts collected from university subreddits to examine the level of distress at the universities [4][25]. A third-party source of publicly available text involves manually analyzing and identifying tweets that inhibit mental health keywords. By analyzing all the research works, they concluded that the lesser the amount of data of the users, the less is the prediction accuracy of the model and more difficult to analyze the patterns of depression. They suggest the works for future research to predict depression from unidentified cases and also to include various social media platforms.

The strengths of paper [4] include the discussion of all the relevant papers that consume linguistic features and the application of natural language processing on the textual data. The strong key point that authors highlighted is the privacy of the users, that is, the information of the users which is used from social media for the research. They discussed that there should be clear guidelines stated [4] between participating entities to ensure the privacy of the users.

On the contrary, Despite the fact that Although many papers which performed prediction based on different criteria were discussed, it lacks the reviews of the papers that consume content other than the textual information to predict the depression. For instance, many pieces of research are being conducted by analyzing the image data and acoustic features which can also be considered.

Nevertheless, another research paper on "Predicting Postpartum Changes in Emotion and Behavior via Social Media" [5], focused on identifying the mental health issues that are dealt by women in their prenatal and postnatal phases [5]. The main motivation behind this paper is to identify the symptoms of the women patients who are already suffering from postpartum depression and help new mothers by making them aware that they are on the verge of depression. There have been effects on the mental health of mothers during the duration of pregnancy and post-pregnancy as there is more responsibility for taking care of the new ones. Munmun and others [5] performed the study by analyzing the activities of new mothers on social media. They include several characteristics like emotion, linguistic style, manner of posting etc. to train their statistical models that will recognize which new mothers will experience some variations in their moods and mental health in their postpartum period.

They proceeded with a dual step approach for getting precise data for analyzing. They began with identifying posts that include wordings related to childbirth and compare it with the announcements mentioned in the newspapers to maintain the authenticity of the data. They extracted various features from the announcement, for instance, sex, age, [5] weight of the infant, state of birth, etc. Secondly, for accurate data, they employed crowdsourcing methodology, and filter out the information that seems to be more

significant, as well as factual chunks of data were investigated to formulate various components of mental state like the emotional state of the mother, utilization of social media by mothers, involvement in postings, communication manner and period. They began with separating the posts of depressed as well as non-depressed mothers and implemented supervised machine learning model Support Vector machines to train on their data. They predicted the characteristics of language used, interaction periods, and sentiments of the mothers more accurately than the ego network [5].

Additionally, the strengths of paper [5] are shown by a detailed analysis of all the factors that should be affecting the cause of depression. In addition to it, all the lexicons and the grammatical structure used by the users in the tweets are also elaborated which is a cumbersome process. The ROC curve in [5] discusses all the fundamentals of the user profile and the comparison marks the significance of each feature used. The research work sets the baseline for future research as the methodology used and the solutions formulated were able to answer many open questions, related to the context of childbirth and the effect of this phase on mothers.

However, there exist some weak points that remains in the discussion, such as whether some medicinal effects are affecting the body of the mother, the kind of medicine consumed by mothers during the phase of pregnancy and the side effects of such types of medicines might be the hidden factor in the cause of depression. There can be lookup for keywords related to the medicines of depression, stress, or other medicines that indirectly affects behavior. This research can be extended by examining the posts on other social networking sites like Instagram and Facebook where people tend to actively post personal life activities along with the photographs, providing authentication as well.

Above and beyond, another research [6] was focused on exploring, evolving the level of openness and exchange of thoughts exhibited by the people in sharing their information over social media platforms. The description of self-disclosure itself states that "the process of making the self-known to others" [6]. The authors utilized the fact that social media is a great tool to speak up, share thoughts and emotions. People are bold enough to talk about every type of issue and concern. The information released by the various users is comprised of various elements like professional experience, personal experiences, health issues, etc. The authors tried to deduce the amount of easiness, experienced by the users in sharing their information and also the acceptance of sensitive information among individuals, particularly related to mental health, by analyzing the data shared by the users, the number of responses and acknowledgments gathered. They used supervised learning techniques in this research.

The paper [6] is an upgraded version of their initial study, in which they explored the potential of social support gathered by people, from the larger social media community. They considered the following factors the basis of their research like revelation by a person about its personality and behavior is believed to ease the person in relieving stress along with depression, as well as social media gives the power to the people which make them to freely discourse regarding the things which bother them. They began their research by collecting the information shared by the users on Reddit. They utilized its API to collect all the necessary information by searching relevant data with the help of keywords related to mental health. They also compiled the data of the users who have nothing related to the vulnerable things and mental health in their posts. For the positive class of users, the ones

which are engaged in actively sharing their secrets and exclusive information, the posts of those users were extracted to validate the authenticity of their dataset.

For making their dataset more genuine, they came up with the manual categorization of the information into three categories: the posts which contain a high amount of self-revelation which means people have openly shared their admissions and more sensitive data, the posts which contain medium amount of self-information, and the posts which contain negligent amount of the information of the person himself. This manual classification helped them to build the dataset for training their model. They conducted tests on the data by application of distinct models like trees, naive Bayes, but the most suitable performer, according to their dataset is the Perceptron algorithm which is a supervised classification technique. Before the application of models, they performed necessary operations on their data, for instance, data cleaning, preprocessing, and feature engineering. The outcomes of the models were enhanced by using the technique of cross validation [6]. The results inferred from the research imply that the high extent of sensitive information has higher chances of receiving responses and feedback, and also within less time frame as compared to other posts.

For the strengths, the paper [6] explored the capability of social media, the platform which provides each person with equal rights, facility to express their feelings and sentiments, as well as how the utilization of social media content can help the detection of several mental health disorders. It also estimates how the user is engaged and dedicated towards the use of social media in sharing personal information, and how they support, along with discussion offered by other users bolsters the person's mindset to share sensitive information with other members of the social media community. Furthermore, it gives the

blueprint of the future research that conjugates mental illness and social media, like estimating the amount of confidence in the information shared by people who are undergoing with mental health issues, how a person can be facilitated with the virtual support, the moment he or she shares something related to depression instances.

To begin with the overview for [7], the users not only post the images but also depict the emotion in the form of captions and hashtags. The authors performed visual analysis on the pictures posted to determine the quantifiable features out of the images [7]. It aimed to answer various research questions by including concepts such as the themes used in the context of mental health posts, the combination of the image, the text posted, and what were the circumstances in which the user has shared such an image.

The methodology of this research includes a collection of large number of datasets. The images were gathered from the social networking site Instagram [7], by using its official API and performing the analysis. The computer vision techniques were employed to address the key findings. For visualizing the features, they used various formats like grayscale [7] image format, in software library OpenCV. They used a two-level approach, firstly to cluster the images based on its pixels to get an opinion about the posts related to mental health and secondly to manually examine the clusters to improvise the result obtained. Latent Dirichlet Allocation [7], the unsupervised learning technique is employed to find out the hidden topics in the textual data, determine the congruency of the image and the written context. The posts shared by the users include various types of images to instantiate, selfies, outdoor places, food images [7], etc.,

For the strength in [7], Ditching the traditional method for finding the symptoms and the sufferers of depression, in a research paper [7], the authors investigated the

evidence of depression using the visual attributes, undertaking the features present in the images shared by the users on their timelines. For depicting the themes manifestations in the posts, they worked on the pixel level decomposition of the image, deep diving into the concept of computer vision. The strength of this paper remains in the elaborate discussion of various kinds of images that can be accompanied by the distressing text.

However, for the weakness in [7], the emotions confronted in this research are limited to five only. This research can be extended by examining the user-level activities by studying various factors like the time of posting, correlation of the image post to its prior and later post. This can enhance the identification of the cases of the users dealing with depression.

The authors initiated the research [8] with the importance of social media data, the widespread increase of mental health disorders, and how the combination of using both the latter, as well as the former can help combat the problem of this disease. The authors aimed to find the elements related to mental health that are passively and actively discussed on social media platforms like Reddit. Their approach included consuming the textual data from the posts published on Reddit. The data incorporating discussion of several mental health problems like Anxiety, Bipolar Disorder, Addiction, Dementia, [8], etc. was filtered out and selected for examination of its properties. They extracted various features exhibited by the text that could depict the semantics used by the users suffering from different problems. The features include the diverse vocabulary used in the context, the writing genre, the kind of mental health keywords used in the posts and comments [8].

The major keynotes for [8] covered that are discovered in this research is the association of the vocabulary used in various Reddit posts that include different health

discussions. In addition to this, the grammatical structure used by the users affected by different problems was also inferred. Their analysis included the inter-relationship of words covering different health domains. They devised already existing approaches to recognize the posts that contain different kinds of sentiments and another algorithm exclusively to detect the emotion of happiness. The uniqueness of this paper includes finding out the emotion of positivity or happiness out of the mental health posts while there is an existing trend that a plethora of health posts are limited to negative sentiment [8]

The gaps of knowledge that are raised from this research [8] show data excluding the context of mental health keywords are not inculcated. This lacks the diversity in the dataset, and the inspection of the inequalities of normal text from all the discussion datasets cannot be formulated. The other point widening the gap to its actual outcome is the omission of comments for examination. The comments can be further exploited to gain more insights into the linguistic features of mental health problems.[8]

The paper [9] provides the details of the characteristics of the users who are on the verge of suicide attempt or who have ever attempted suicide [9]. They utilized the user data released on social media platforms to analyze their characteristics, predicted significant factors and symptoms that are usually expressed by people in the context of suicide. Also, they implemented statistical machine learning [9] model for further enhancement of their research. Their methodology includes collecting data of the users, who have already drafted suicidal attempts in their posts, from their specific accounts to maintain authenticity.

For the strength in [9], the point which makes this study more advanced is the collection of user data before their suicidal attempt, which distinguishes the user behavior in both situations. Moreover, instead of the traditional preprocessing techniques [9], they

abstain from data stemming and lemmatization to know the notion of tense used by the user to convey its emotion. The findings unearthed during their study include the sentiments of the user near their suicidal attempt, the semantics used by the user, and the incorporation of other emotions [9] by the user. Afterward, they used an n-gram approach to deal with the linguistic properties of the text [9]. The depiction of several sentiments behind the intention to kill oneself include antipathy, loathe sorrow, acrimony, and the findings highlighted the abundance of such emotions in the tweets by the user. These similar issues are also addressed in the study of depression.

However, for the weakness of [9], this paper has not managed to address the key issue, depression and thorough study of these emotions, it can be inferred that continuous and repeated suffering from these emotions, led to depression, and thereby, leading to suicidal situations. Secondly, the research is restricted to the emotions and the number of tweets posted by the user. The research can be extended by obtaining the vocabulary used in the context of the suicide which can help identify the candidates before the attempt.

Moreover, the research [10] exclusively aimed to appraise the potential of social media data released by the users on the digital platform and assess the intensity of depression, of the various users affected. Instead of various approaches followed, this paper essentially addresses the user-level characteristics. The paper targeted to unbox all the direct and indirect factors and hints that can be extracted from the information shared to study the human-nature during the phase of depression. The procedure of the data gathering was referred from the initial paper[1], but respondents who participated in this study were Japanese and were asked to take two tests BDI test[10] similar to CES-D[10] questionnaire to evaluate the level of the affected users. The next procedure involves deriving features

that inhibit linguistic properties and finding the correlation existing between different words.

The strengths of paper [10] include the in-depth analysis of the user behavior and the following features were obtained from the communications [10] such as tweet frequency, hour, following, follower, URL and mentions. The topic feature was obtained by applying Latent Dirichlet Allocation (LDA), to find out the probability of each tweet concerning its topic. The boxplots in the paper [10] indicate the different parameters that classify between depressed and non-depressed class. The model for separation of depressed and non-depressed class induces a supervised learning technique. A support vector machine algorithm along with 10-fold cross-validation techniques are used, and their model predicts the output with 66% accuracy and 61% precision [10]. The authors were able to justify their research with their outputs. The feature engineering and feature extraction paved the way for future research that can be modified for later use.

Besides for the weakness in [10], the findings of the paper regarding the language used in negative sentiments and depressive patients, and the effect on model with the variation in size of data is interesting, however, this research can be extended by examining the tweets of the user and identify the causes of an earlier stage of depression. Additionally, the participation of more users can be encouraged from different countries, to visualize the pattern and characteristics of the depression affected social media users globally.

To begin with, another research paper contributing towards a topic on “Mental Health Computing via Harvesting Social Media Data” by Jia Jia [17] helps to understand a need to develop a system for analyzing and maintaining a theme to control the problems related with mental health. In their research, they begin with the need and motivation which

targets to detect issues related to the causes as well as explain that some of the treatments still depend upon the stage of the history of a system [17]. They focus on social media as a platform over which data is growing daily to categorize their research on two major problems termed as depression and stress [17].

They began exploring various new ideas for data collection which involves several platforms such as Sina Weibo, Tencent Weibo as well as twitter that helps to collaborate a multi-cultural information source to break the chances of ambiguity while applying models. Such types of diverse elements help to break the ground reality of applying a huge amount of time using the existing datasets and not exploring much more unique or novel ideas to solve or reveal some new research questions. Nevertheless, to compare the extraction process, a real-time dataset brings unprocessed raw information hiding several secrets whereas every preprocessed dataset that has been cleaned already gets used in several analysis and shows no progressive statistics, which makes raw dataset helps better to portray a baseline of hundreds of ideas for future work to be done upon them.

For the detection of stress-related information, they used extraction of tweets to build up various datasets ranging from DB1 to DB7 under various platforms such as Sina Weibo, Tencent Weibo and Twitter that holds different labels, different number of users as well as different number of tweets [17] which are interlinked to a dataset DB2 [17] that holds a set of recorded scores based on different cognitive mental stress levels to cross-check the listings that were carried away or moved to different datasets in their model. They applied various demographic attributes to the model for defining various categories that will help to extract datasets holding up contrasting features such as language, ocular imaged or

visualized content, communal behavior, and reciprocal actions while communicating on social media[17].

They used CAE termed as a cross autoencoder that helps to decrease the noise instances and arrange the data dimensionally and then applied a loop contortion neural network in a solitary single Dimension [17]. To add to it, after going through a series of tests to validate and verify the model output on an F1-scale attains 93.40% [17].

For the strengths in paper [17], the research helps to understand a lot of future work that could be done by the use of separating, classifying different set of tweets from diverse platforms and then contrasting each category by labeling them as stressed, non-stressed or depressed and non-depressed, as well as instances related with an element that causes stress termed as stressor in their research[17]. Moreover, to amplify the execution they set up a Multiple task learning model [17] that helps to bolster the output by comparing congruency encompassed by each task and create a modulate balance improvising CNN.

To add to it, they engage the SRRS termed as the "Social Readjustment Rating Scale" [17], which helps to achieve excelling results by projecting stressor instance and equivalent stress levels [17]. For conducting their study, and by following the standards by ICD-10[WHO,1992] [17], they sampled data user's data for nearly a month. To maintain the dataset generic, they scanned real-time tweets which help to maintain the authenticity of the dataset. Besides the derivation of user profile features, they also acquired features related to the topic as well as domain. While preprocessing the data, they recognized several overlapping features and features relevant to specific dataset, and came up with the concept of isomerism[17], in which feature can take different form in different spheres and divergence[17], in which feature can have contrasting values in different spheres. They

pursued their processing by applying the Deep Neural Network model with Feature Adaptive Transformation & Combination strategy (DNN-FATC) [17] to deal with the complications of isomerism and divergence. They handled isomerism by transforming and normalizing the features to stabilize the distribution of data, and then applied deep learning models to regulate the divergence.

By the consolidation of different features, they applied distinguished DNN models and accomplished miraculous results by using this approach. Furthermore, they disclosed many factors like the affinity of the emotional state of both kinds of users, stressed and depressed, the language characteristics, time of their tweet postings, the interaction of users with other users, association and connectivity with other people, and inspecting their social circle by performing analysis on their data. For the prospective work, they planned to inculcate offline data resources to make predictions more relevant. Having a different ideology to deal with the tweet data, this paper plays a prominent role in improving further research.

Another research paper published in the year 2017 at International conference on Advances in Social Networks Analysis and Mining by Y. Amir et al[18] contributed their work towards the topic “Semi-Supervised Approach to Monitoring Clinical Depressive Symptoms in Social Media”[18]. As the involvement of users has been expanding on a large scale to clear out their emotions or perception towards daily work and efforts in the form of postings on social media [18]. Their research contributes to the detection of symptoms that imply depression in users. Their research works opposite to contrast with the traditional methods that use survey forms acknowledging the individual already getting awareness of being open to being diagnosed through the study investigating symptoms

indicating depression. Whereas, in their research confidentiality is being considered as a key objective to derive tweets from users that have accounted to create self-posts on social media indicating symptoms towards a manifestation of having depression [18].

In this research, they proposed a semi-supervised statistical model [18] which imitate a common methodology formed by coordinate matching of a recommendation of outputs with the findings from patient health questionnaire -9 (PHQ-9) which is used by therapist as a survey to scale a type and nature of depression or stress level in an individual. They balanced an equation of the set of output lines from their model in terms of word handling arrangement and predilection of topics with the results concluded by the PHQ-9[18].

This approach helps them to yield an accuracy and precision of 68% and 72 % respectively by their automated filtering model in detecting any evidence indicating of having depression. The action of their system follows the Diagnostic and Statistical Manual of Mental Disorders (DSM)[18] which concludes that an individual could be analyzed to be in a state of depression for having signs indicating depressive symptoms persisting for a certain period [18]. Some of such symptoms from this research were measured to compute the seriousness of the depression state of an individual.

The indicators were based on following signs for example sense of diminishing amusement, satisfaction, disheartened, demoralized nature, difficulty in sleep, drop off stamina and strength, loss of desire for food leading to improper diet, feeling futile, Lack of interest leading to feeling unproductive or fruitless, Losing focus, getting lethal thoughts to kill oneself along with many others [18]. They followed a Latent Dirichlet Allocation (LDA) [18] model for defining a set of symptoms that comes under the category of particular topics of symptoms based upon users helping in collecting data set information.

To add to it, they also followed a (ssToT) approach termed as semi-supervised topic modeling over time which helps to learn from the auto arranged topics in categories defined by symptoms based upon the indicators defined above forming a cluster of dataset values into refined information.

Moving further with another research carried on the topic “Monitoring Tweets for Depression to Detect At-risk Users”[19] by Zunaira Jamil et al[19] which focuses on a big set of data launched by Bell Canada in the form of a campaign launched in 2015 which was based upon postings with the hashtag #BellLetsTalk[19][97]. In this campaign, the main target was to record tweets by users revealing their mental wellness and psychological health-related issues with the medium of social media [19]. This was a successful launch of huge dataset information leading to an acknowledgment of 122 million tweets and other activities on the social media platforms [19]. This research contributes towards creating a classifier based upon user-level architecture because such an enormous number of datasets takes a large amount of time consumption for preprocessing, cleaning and other actions to perform further analysis. Therefore, they decided to break down the structure of the dataset into user concentrated parts based upon all the tweets by users who were Canadian. Moreover, they also label a tweet-level structure which contrasts some features from the user level structure and has differences in terms of dataset values, as well as number of users according to its type of user or tweet level.

Their research inclined towards the prediction of depression by identifying users who are having a higher exposure to danger with a better precision and recall value than to the accuracy score [19]. In the tweet-level structure, they used prediction modeling with a motive to check whether a tweet has more compatibility to detect depression which leads

to a problem of biased results showing 5% tweets to be depressed and other 95% to be non-depressed. Furthermore, to control and balance the set of equations in the dataset with the model they used undersampling[19] which leads to score a much better recall in contrast with the precision. The main approach was set up using an RStudio IDE (integrated development environment) [19] under which firstly, the original dataset was trained by Linear SVM. Secondly, another set of data stabilized by applying SMOTE was trained using the same procedure of linear SVM. Third and the last training was carried out on a dataset that was counterbalanced by applying to an undersample to form an equitable dataset[19].The model at tweet-level shows a recall value of 0.80 and accuracy for 61% at an undersampled data set value whereas the user-level for 160 users on original dataset scores an accuracy of 78.72% and recall value of 85% which is comparatively higher than other experiments carried in this research[19].

To begin with, another interesting [20] paper which expounds about problems related to mental health care contributes to learning more with research on the topic “Detecting Signs of Depression in Tweets in Spanish: Behavioral and Linguistic Analysis” [20]. As Explained by the topic, their research effectively concentrates on the content available on social media platforms generated in the Spanish Language. Furthermore, they did a good comparison analysis in their literature review as a lot of models have been already implemented to explore more with the use of sentiment analysis which grows into a category under mental health care for predicting depression. In this research, they worked on collection of tweets from various words derived in a direct manner which are pertinent to depression and developed balanced model by making a manual as well as an automatic selection of users through which they came up with an advanced subdivision category of

two types of datasets based upon tweets indicating a depressive behavior in contrast with a dataset which carries users showing depressive nature. To add it to it, on the other hand, they also collected tweets using Spanish stop words which holds up a sum of random users that have not mentioned any information related to mental or depressive symptoms just to develop a counterbalanced dataset which they referred to as a control dataset[20]. Moreover, only a few of the researchers worked on population targets by language barriers such as this research paper that ultimately results in a progressive model based on Natural Language Processing NLP, Part-of-Speech (POS) and Lexicon Analysis [20].

For the Strengths, their paper looked for a valid area to be worked upon which attracts analysis for information in Spanish because a large number of users get missed or skipped because of Specific lexemic content-oriented data Which is based upon a particular language [20]. This paper helps to target places where most of the information is dealt in Spanish which helps us to understand and reveal a new work culture along with some other demographics based on language due to which any new depression symptoms get identified.

On the other hand, the weakness for [20] becomes that they did not have a verified version of a model under which they could scientifically and medically prove that the result of the investigation done through the model for prediction of depression is generic. It is just based upon the outcomes of the algorithmic frameworks performed on the datasets that were based upon information available from tweets by users publicly. Moreover, there are chances of ambiguity and discrepancy between the type of emotions, feelings, selection of words, as well as interpretation of the user which could lead to a false outcome by not matching keywords with the bag of words entangled. Also, some unsolved research

questions target the partisan situation occurring due to private or spurious account users that could influence datasets with depressive tweets or by several users in depression.

The paper [21] focused on identifying depression among individuals by exploring and exploiting social media posts of different users. The authors extracted data from various social networking sites such as Facebook, LiveJournal, Twitter, etc. After the extraction of data, their methodology includes the application of several functions on data in a certain order. They began with feature selection and feature engineering which means to remove unnecessary columns existed in the dataset along with converting the text to other forms of variables that are compatible with the prediction algorithms. Thereafter, they performed data cleaning and preprocessing, to remove stop words, altered case of the text, transforming from high case to lowercase, executed stemming, tokenization and then split into the training set, as well as testing data set before the application of the model.

They took an equal amount of depressed and non-depressed posts, and then labelled the training dataset manually. They used two algorithms for the prediction Support Vector Machines and Naive Bayes, both of which are supervised learning approaches. They evaluated the performance of their model by calculating various metrics such as precision, recall, and accuracy. They compared the performance of their algorithm with the prior research conducted and discovered their research to be the lower ones.

For Strengths in [21], the paper consumes data from different social networking sites, therefore their model is not dependent on one source or similar type of data. They considered the balanced dataset for their prediction, an equal amount of depressed and non-depressed posts to reduce the effect of bias. The data cleaning procedures and the operators [21] methodology is effective in dealing with such type of textual information.

Furthermore, for the evaluation of their model, they considered actual patients and processed their responses to verify with their model.

Nevertheless, for weakness in [21], the algorithms, Naive Bayes and Support Vector Machines failed to perform more accurately with high rates of recall. Secondly, for verifying the prediction, the number of test cases used is significantly a small amount, 30, which is not enough to generalize the model as well as results. The description of such users, whose tests were conducted to validate the performance of their model, was not elaborated. There should be an explanation, by which means the users were selected and communicated. Moreover, user history was not considered when exploring the social media account of the users. Therefore, patients suffering from chronic depression cannot be identified.

A paper published in the journal of medical internet research in 2017 examines for Systematic review on "Researching Mental Health Disorders in the Era of Social Media" [22]. This paper focused on scrutinizing the perception of the approach used by models towards analyzing and predicting outcomes of having symptoms indicating depression in users within social media structure. They reflect on the importance of ethical approval and moral interest of the users as well as delve into finding limitations related to the modern techniques used by the researchers. They find articles 7 years of articles from 2010 to 2017 [22] which are related to symptoms indicating depression issues concerned with mental fitness as well as the psychological well-being of users on social media. Their research is a more detailed architecture of comparison between all these articles published under medical and health as well as computer technologically associated journals [22].

For Strength in [22], this type of research helps us to learn about the discrepancies and working layout using the (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) PRISMA [22] model as well as the comparison of all the technologies used in the past within the current methods. Such a study helps us to learn about various limitations, lack of inputs and techniques which could be taken from one research to another because of their cross-connection being carried on a single platform within the handling of 5386 articles [22]. Their research strongly worked upon a background study to eliminate and came with an idea of having a model that proves the best technically, as well as ethically for future analysis for predicting depression.

Additionally, for the weakness [22], they should compare their research outcomes with a paper that has derived from reviewing all the other sources using journals, articles, and blogs on topics related to mental illness to strongly validate the objective. Moreover, they selected a smaller number of articles from the total extracted which could manipulate the resulting outcome for the training model to counterbalance their established selective keywords.

A paper by Victor et al [23] on the theme “Twitter: A good place to Detect Health Conditions” explores that within the expansion of the social networking and microblogging websites, twitter scores a high affinity towards publicly available information. This Indicates towards increase in various websites that are related with Mental illness and health care over which users can post their queries as blogs, For example, Twitter uses retweets as a reply or a new post by a new user which ultimately spreads information by a two-way interaction between few individuals to be publicized over and over helping others suffering from a similar condition. In their research, they extract tweets by setting up a set

of interpretations of statements and linguistic verbalization using machine learning[23].To add to it, they focused on a specific set of features within compatibility to run time in completion to boost outcome. Moving further for testing, the model was performed in action using 4 predefined circumstances [23] such as flu, depression, pregnancy affliction as well as eating ailment [23] with a combined structure-based over this information collected from Portugal and Spain[23].

Whilst for the Strength in [23], they scored high values in various types of criteria followed by both categories of feature selection and results which were carried out without feature selection [23]. They made a very easy understanding table distribution for comparing the gain between the average results compiled based on Precision, Recall, AUC as well as F-Measure [23]

Whereas for Weakness in [23], they could also consider various other details other than the four categories on which they put the keyword matching for measuring symptoms such as relationships, emotional stress, professional as well as personal category. Moreover, they could also consider more locations to collect information extracting datasets as testing data and make a verification model for further analysis using their proposed model as a validation criterion.

Moving further towards a very interesting paper by Natalie Berry et al[26] revolves around research which helps to understand the logic behind users interacting with social media as a platform to articulate their feelings, Opinions, Health Issues as well as Feedbacks, etc. To dive deep into the details of this concept, they published an article on the topic “#WhyWeTweetMH: Understanding Why People Use Twitter to Discuss Mental Health Problems” [26]. They adopt a peculiar hashtag dependent probe forming an

idiosyncratic approach for extracting tweets utilizing the Twitter Streaming API as well as the Twitter search API [26]. They broadcast the hashtag #WhyWeTweetMH from September 2015 to November 2015 and accrue the dispersed extracted tweets piled up into a dataset for further analysis.

Nevertheless, for the strength in [26], Their research was highly invigorating to demonstrate curative advancements for self-rejuvenation of mindset for the users to accept and open up socially with the use of such hashtags related to mental health [26]. They worked on a theme-based structure that shows social involvement and connectivity in society, to aware users for such mental health-related issues, to provide a safe zone for letting users open up and reveal their issues with a sense of security along with acknowledgment [26].

The weakness of this research paper [26] was revealed under the collection of dataset strategies which should be given more months with better advertising of the hashtag used for research. Moreover, they collected a very small amount of data for which ethics could be an issue in the future because users' identity gets transparent when they use such specific hashtags and tweets are easily traceable.

Another interesting paper by Mowery. D et al, on "Understanding Depressive Symptoms and Psychosocial Stressors on Twitter: A Corpus-Based Study" [27] concerns to develop a model for a deep understanding of major depression problems that cover nearly 1/6th of the total population in the US [27]. Depression is culpable for being the 5th colossal reason for mental health diseases and various other psychological problems in the US. They created a method that helps to utilize the Diagnostic and Statistical Manual of Mental Disorders (DSM-4) as well as (DSM-5) into an explanatory manner modifying the

body of text called the SAD corpus in their research. They begin with the extraction of 9300 tweets out of which when the analysis has performed the outcomes from their model reveals that nearly 72% of tweets were asymptomatic.

Into the bargain for the Strength in [27], They show traces for non-relevant context by nearly 72% of tweets which contains the word depression but on any particular situation and not on a real feeling based upon a human being helping to remove biased results in the future. They worked to form a model that helps to understand the real meaning of the keywords utilized in a comment accounted by similar tweets that have the same words but different meanings.

Quite the reverse in the case of Weakness for [27], The research paper utilizes a very low extraction of tweets and did not apply any meta-learning modeling on the further description of a better understanding of depressed users such as a graphical or visual representation. The paper lacks some information on the topic of demographics which could relate to a better understanding of attributes and feature engineering with the types along with methodology of algorithms in more detail.

Elaborating other research paper [63], the paper aimed to diagnose the symptoms of depression using tweets posted on Twitter. They collected tweets from the users and segregated into three categories, control, depressed, and PTSD labelled user [63]. The users were split into testing and training samples, and their vocabulary features were extracted. These features were cleaned by performing extensive data cleaning and altering them by applying feature engineering. Word embeddings were obtained from the words and were optimized, as well as encoded with vectors. These features were passed onto deep learning frameworks and passed on to different architectures of neural networks for the prediction

detection. For the validation and performance of their approach, they performed experimental analysis on two different datasets. The use of deep learning framework, one of the advanced frameworks for testing the depression detection among textual features is significant contribution and strength of this research [63]. The main weakness of this paper is it has not combined other features of the user activities and their behavior on social media in order to depict the depression severity among individuals.

Describing about another research paper, [64], the purpose of the research aimed to diagnose any characteristics or the symptoms that could indicate the signs of depression among users present on social networking site. The data resource for their research was reddit website, from which users' posts were collected. They performed data cleaning using Natural Language Processing toolkit [64] and worked on assortment of features by application of various natural language processing techniques. Application of topic modeling aided to gather latent features of the textual information. They employed supervised machine learning algorithms to appraise the performance of their algorithm and conducted experiments by combination of single as well as multiple features.

The strengths of this paper [64], includes the application of multi-layer perceptron indicating the application of deep learning techniques. Finding the latent features of the text using LDA also enhances the understandability of the context. However, the research is confined to the semantic features obtained from LIWC [64], and the topic modeling features. The lexical analysis of the posts could yield the vocabulary features that could help in better detection of depression affected patients.

2.2 Comparative Analysis

This section explicates the comparison between all research papers which contribute towards a similar motive for predicting depression. To understand the prime factors by developing a criterion based upon topics including: Research Motive, Dataset collection, Dataset features, Methodology, Results, Strengths, as well as Weakness.

The tables 1 shows a tabular formation of reviews carried on most relevant research papers related to this study. The key aspects of research [1] focuses on different attributes such as emotion, linguistic style and engagement of the user to predict the depression levels among social media users. On the other hand, the research [2] tried to find the depression level among social media users by quantifying it to a numeric value called Social Media Depression Index. However, the number of user records in research [2] are not enough to generalize the model results on whole population. Further, the research [5] focus to identify the symptoms of women patients who are already suffering from postpartum depression and help new mothers by making people aware of them. They verified their model by retraining small sample of the dataset in order to see similar results. The research [6] after stemming and preprocessing of data collected, implemented the n-gram approach on the dataset. They also included length of the post, as well as the category of the person who posted. The research [6] score better accuracy than the research [1], [2] and [5]. The research [7] focuses on identifying symptoms of depression using visual attributes from Instagram posts or images posted by the affected users. They extracted color profiles and built grayscale histograms, noted out brightness, saturation, as well as contrast image distribution using OpenCV. The research [7] applied a different approach to use visual attributes instead of using only text and worked on compiling the visual features with the

categories of the images. The authors concluded that anxiety is the highest emotion [7], followed by anger when expressed through images. The research [8] focus to define the linguistic characteristics of users affected by mental health disorders. They constructed a parse trees to determine the syntactic structure of the sentence [8]. They mapped happiness and sentiment scores by using dictionaries. Another research [10] focus on estimating the degree of depression by exploiting the social media activities of the user. The key feature of this research was construction of a website to ensure the authenticity of the affected patients [10]. On the other hand, the research [17] worked on accumulation of social media data to detect and predict mental wellness of the user. They combine the linguistic and visual attributes of the posts on twitter to develop a multimodal dictionary learning model. Nevertheless, the research [18] implemented various semi-supervised models to monitor clinical depressive symptoms in social media. They classified tweets into several buckets for validation of their approach and compared all of their model's performance in each bucket. The research [19] aim to detect the users who are on the verge of depression from their social media activity. They used combination of LIWC features and communication features of the user. Their SVM model achieves an accuracy of 78.72%, which is higher than the SVM accuracy scores in research [1], [2] and [5]. The research [63] contains the tweets of various users labeled as Control, depressed and PTSD affected patients. They converted text to word embeddings and fed data into neural networks, scoring 80.5% accuracy and 83.8 % recall. Lastly, the research [64] used the dataset created by Inna Pirina [64], containing depression indicative and standard posts. They prepared N-gram, Linguistic, as well as topic modeling features on text level classification and received 91% accuracy and 93% F1-score which is highest than all other research papers in table 1.

Table 1 Showing comparison between different research papers

Research	Predicted variable	Dataset collection	Methodology	Results
[1]	Depression levels	Crowdsourcing (476 users with 2 million tweets)	Support vector machine (SVM)	70% accuracy and 74% precision
[2]	Social media depression index	Crowdsourcing (69k tweets)	SVM with rbf kernel	73% accuracy and 82% precision
[5]	Postpartum depression	376 new mothers with 40k twitter posts	Support vector machine (SVM)	71.11% accuracy, 74% recall and 72.9% precision
[6]	Mental health related self-disclosure	4 million Reddit comments	K-NN, Decision trees, Naive Bayes and Perceptron	78.4% accuracy, 86.9% recall and 74% precision
[7]	Visual attributes Of mental health disclosures	Extraction using Instagram API	SURF, LIWC and LDA	Visual features and themes
[8]	Linguistic characteristics of mental health	3.97 million reddit Posts	Parse trees, NLP, word-based classification	Observed variations in linguistic features and vocabulary
[10]	Estimating degree of active depression	Survey responses and tweets of 209 participants	Support vector machine (SVM)	66% accuracy
[17]	Computing mental wellness	Extracted tweets using Twitter, Sina Weibo and Tencent Weibo	Multimodal dictionary learning (MDL)	F1 score of 85%
[18]	Clinical depressive symptoms	2000 self-reported depressed users and 2000 random users	LDA, LSA, BTM, P-LDA, K-Mean, Naive Bayes, SVM and ssToT	68% accuracy and 72% precision
[19]	Detecting users on Verge of depression	Bell let's Talk dataset	LIWC, SVM, NRC sentiment, emoticon and Readability features	78.72% accuracy and 70.83% precision
[63]	Depression detection of Twitter users	Clpsych2015 and Bell let's Talk dataset	Word embeddings, CNN, RNN, Bilstm	80.5 % accuracy and 83.8 % recall
[64]	Detect depression related posts using Reddit	1293 depression indicative posts and 548 standard posts	N-gram, LDA, LIWC, SVM, Logistic Regression, Random Forest, AdaBoost and Multilayer Perceptron	91% accuracy and 93% F1-score

Chapter 3: Dataset Review, Analysis and Dataset Details

The following chapter highlights the investigations carried on technicalities related with various publicly available previous datasets that contribute towards predictive analysis of any one of the topics in context to mental health related with sentiments, depression as well as user behaviour and activities. To add to it, this chapter describes the origin, methodology of collection, strengths as well as weaknesses of available datasets. Further, the dataset analysis is performed in order to portray an easier understanding of all the past datasets available in a tabular form. Lastly, details of the dataset for this research, utilized under experiments as well as the proposed solution are covered.

3.1 Dataset Review - Survey of Existing and Related Datasets

The dataset Sentiment140 [34] is a publicly available dataset that is meant for academic research in the domain of NLP, and social media analytics. The methodology of its extraction is discussed in the paper [35], in which all its attributes were discussed. The dataset consists of the tweets done by the users on the social media site Twitter [36]. It is specifically built and designed for studying the sentiments of the users.

The attributes in the dataset include the id of tweet, indicating unique number assigned to the tweet, date of the tweet posted, the query information present in the dataset, the user id which identifies each user, the text of tweet and the polarity[35] of tweet, which contains the numeric value which classifies the positive and negative tweet. The total number of tweets present is 1.6 million. Other characteristics of the data included are the average length of the tweet which is 78 words [35], and the URL cited by the users that are present

inside the tweeted text. Other deductions that were formulated from the dataset include the repeated characters, the different domains discussed by the users, etc.

For strengths, the dataset [34] sets the baseline for studying the emotional patterns described by the users on social media platforms. It contains the ample features necessary for the sentiment analysis in various domains. Secondly, the size of the dataset is large enough to yield substantial results for the sentiment prediction.

However, for weakness in [34], it lacks neutral sentiment, the research is only limited to positive and negative sentiments. The dataset contains insufficient information in context to the users which cannot be suitable for studying the user aspects while writing their emotions. The dataset contains instances of diverse fields that do not make it suitable for research in the domain of mental health and depression. Instead, relating instances can be filtered out for use in the domain of health.

The NRC Emotion Lexicon [37] is a sequence of English words and their relationship with eight basic emotions (anticipation, trust, fear, anger, surprise, joy, disgust, and sadness) along with the other two sentiments (negative and positive) [37]. The markings of the datasets and segregation were manually done by crowdsourcing. The dataset can be used for commercial and non-commercial purposes. The dataset was built in the year 2010 and contains around 15k number of terms in the form of unigrams, as well as around 25k of senses indicated by the users. The categorization of the association of words was divided into three quantities depending on the intensity of its conjunction with the meaning.

For strength, the dataset [37] provides the basis of sentiment analysis and has been used for studying the emotions depicted by comparing its meaning. The study has helped devise the machine learning model as the pre-formulated list can be used, thus reducing the

manual load. For the weakness in [37], though the annotation of words has been done manually, there is still a chance of human error while determining the polarity of the word and also in the cases of the ambiguity. More sources of vocabulary can be compiled to generate a more diverse list of emotions.

Another dataset, derived from the social media platform is the Reddit Self-reported Depression Diagnosis [38] dataset which includes the Reddit posts, containing information about the users who have proclaimed themselves to be suffering from depression. The depression affected users are approximately 9000 in number and around 107,000 control users that were having an intersection with them. The dataset contains only the public posts and is not available under the open-source license due to privacy concerns as it contains sensitive information regarding users. The extraction process involves the hardbound rules such that, the users who have posted such a scenario "I was diagnosed with" and must have around 100 postings [38] were included for involvement in the dataset. The dataset was further split into training and testing datasets.

Moving to strength, the dataset [38] was crawled from the social media website reddit[39] and has utilized the communication threads to consume the user data. The methodology can be used for extracting information present on various social media platforms and paved the way for the utilization of streaming information for the health domains.

Nevertheless, for weakness in [38], there might be cases that could have remained undetected by the hardbound rules of extraction, that only the self-proclaimed user will be considered for research. The users who have used depression-related vocabulary loosely remain undetected. Secondly, the authenticity of the posts cannot be verified whether the

user is genuinely affected by depression. The size of the dataset is not large enough to represent the whole population suffering from depression or could correctly detect depression-related symptoms.

Other psychology datasets include [40], Catell's 16 Personality Factors test dataset that was formulated based on the 16 factors based on the study of answers to several questionnaires. The total attributes are 167 in number which contains gender, country, age, accuracy [40], and answers 163 attributes containing the answers to the personality questions[41].

Moreover, for strength in [40], the questionnaire has helped assess the personality of the person. The role of this dataset [40] in the context of depression, is the detection of stress by studying its responses to the various questions. The main foundation of this dataset is to perceive the idea of the existence of a correlation between words while depicting the depression or sometimes the personality factors that can aid in the detection of the mental health status of the person.

Oppositely, for the weakness in [40], this data contains self-responses by the users, which can lead to overestimation and underestimation of some factors by the user itself. To determine the depressed user precisely, there is ambiguity in deciding which factor or the combination of the factors are more robust to indicate the overall mental strength and constitutes the overall personality of the person.

Another dataset that exclusively stresses the problem of depression is the Depression Anxiety Stress Scales [42]. The dataset not only aimed to detect the problem of depression but also focused on its additional effects like stress, anxiety. The questionnaire consists record of 42 symptoms related with Depression, Anxiety, and Stress [42]. The whole of

the attributes assesses the level of depression present in the users. The questionnaire for accessing the factors along with responses is available at [43] and includes all the mental behavior contexts in different situations. The scale of the depression assesses hopelessness, useless life, dysphoria, low self-esteem, lacking in interest and activities. The scale of anxiety assesses arousal, muscle movements, anxiety on different occasions. The scale of stress assesses relaxation, arousal of feelings, and impatience. Users are asked to rate the questionnaire on a 4-point scale.

For strength, the dataset [42] is related to the research in the context of depression to identify the new users who have not been detected as depression affected individuals yet. This can also be used to find the association of the sentiments that belong to the different categories of users like depressed, non-depressed, patients suffering from anxiety, and stress. Whereas for weakness, this dataset [42] also contains self-responses which does not validate the authenticity of the responses and the parameters. The questionnaire does not contain the hardbound questions like the prior knowledge of the user, the medications prescribed to the user and the activities performed by the user that could indicate the severity of the question.

Another dataset [44] that is constructed by the collection of the tweets from the years 2009 and 2016 to study the users affected by depression. The dataset comprises three categories, the users who are severely affected by depression, another category in which users have not disclosed their feelings related to depression or stress and another category in which users have loosely mentioned keywords related to tension and stress. The third category of this dataset is the control dataset used in [16].

Additionally, for strength, the datasets have an enormous number of recorded users and their tweets to study the spread of depression in streaming online data. The dataset is well labeled in terms of identifying users who have been diagnosed with depression. There is a robust check to recognize genuine users rather than relying on the probability measures to estimate the depression effect.

On the flip side, the large dataset requires high computation to produce the output. Therefore, to deal with the preprocessing methodologies and preprocessing, it requires some additional source of processing to produce the output in optimal time.

Another dataset that consumes the acoustic features to detect depression is the DIAC WOZ Database available under [45]. The audio recordings were provided by DIAC WOZ [45], however the linking with the depression scales along with the association was completed by USC's Institute of Creative Technologies and was launched as the subpart for the Emotional Challenge Workshop held in 2016. The dataset [45] includes around 190 sessions with an average length of 15 minutes. The participant was asked to take the PHQ index test to determine the personality score and its score whether he is affected by depression or not was determined. Afterward, the session was conducted in an innovative way, in which the participant was asked questions about the personality and behavior of the virtual interviewer. The sound exhibits different patterns like noise, silence, and its features can be derived to study the mental state of the person.

Even more for the strength, the dataset [45] was collected using audio and video processing techniques. The involvement of the human interviewers and human-controlled agents led to the capturing of the hidden aspects of the user which may not be identifiable over just the textual information or the audio information alone. The factors like voice

quality[45], dialogue annotation outstands the methodology of the collection of other datasets used in depression detection.

Whereas, for the weakness in [45], the collection of such types of datasets is an extensive and rigorous task, as well as requires some financial resources or other means to capture the data and annotate it. The virtual interviewer [45] used in the collection of the responses is a controlled agent and may not be aware of all the human emotions which could make users uncomfortable or nervous to some extent.

The other dataset that utilizes wearable technologies to identify depression is the Depression dataset[46]. Wearable gadgets are accustomed to recording various physical activities of the user and hence, record various health indicators like pulse rate, heartbeat, calorie intake, calories burned, daily physical activity, etc. The dataset[46] contains the sensor data of different users collected according to their physical activities. Data collection methodology includes actigraph [47] which measures activity levels, concurrent frequency and voltage were acquired. Two folders are contained in the dataset, one folder acts as the control group, from which affected users will be compared.

The activities obtained from the actigraph are compiled in CSV format. The columns include timestamp [46] (time recorded in minute intervals), the date at which the activity was recorded, and the activity performed by the user. The MADRS [46] scores were also formulated based on this dataset which contains the following attributes: patient number which uniquely identifies patients, duration for which activities were recorded in the form of days, sex of the patient, age in the form of number, marital status of the patient in the datatype string, the education level completed by the patient and the type of depression with which the user is affected. The dataset essentially aimed for applications where

statistical modeling can be applied to study the pattern of depression and secondly, to analyze how the daily routine of a person can make him a victim of depression.

On top of that for strength in [46], The dataset is constructed using the output of user activities captured through various sensors. The uniqueness of this dataset is that it captures the real-time data of the patients that are suffering from depression therefore, it helps to analyze the effect of depression on physical activities. This dataset can be used to infer the sleeping patterns of the patients, the amount of laborious and exhausting tasks done by the patients as compared to the normal people. Moreover, it contains the records of the profession of the participants which helps to associate the activities of the patients accurately.

However, for weakness in [46], The number of patients and control users is not enough to generalize the findings. The dataset provides the intensity of the activity but not the exact activity of the user. This can lead to vague interpretations of the control user, for instance, not involving in physical activity but might be mentally more active. Therefore, more strict supervision is required to capture the readings.

Another dataset[48] that contains the mental health conditions of the adults of California. Data of only those users were recorded who has ever suffered from the distressing disorder. The data is collected by conducting a telephonic survey in which their health-related disorders, chronic diseases, behavior characteristics, and mental health attributes were recorded.

For strength in [48], All the major indicators of depression-like diagnosis of chronic diseases, health and behavior conditions are recorded which is vital for detecting the

depression along with its cause. Different parameters of the users within different age years are also contemplated to understand the variability of patients affected by depression.

Conversely, for the weak links in [48], The data does not contain any gestures or the actual input from the user that could aid the use of this dataset in the prediction of depression. This dataset is limited for descriptive analytics, to understand the current scenario, given the telephonic responses are valid and authenticated but it cannot be used for identifying the new users who are at the risk of depression.

The other dataset[49] collected by scraping the twitter data that contains the keywords related to depression, anxiety, stress, suicide, guilt, etc. and other mental conditions. The programming language used for scraping the tweets of the user is Python and it uses Twitter Streaming API to construct its database. The data is collected in JSON format and needs to be preprocessed for analysis. The dataset consists of approximately 11k rows and around 40 attributes describing the context of the tweet. The attributes contain the user id, tweet id, username, the information of the user, the number of followers, number of retweets user has got, the participation of the user in posting the tweets, the time during which user was active, hashtags, user mentions, demographic information, etc. Likewise, for strength in [49], this dataset provides a substantial amount of necessary information to find the depression-related linguistic features and the vocabulary used that exists among social media users.

This dataset can also include the time zone and timestamp features that can find out the latest features of the depressed users. These features can be engineered in several ways to yield results, which can be useful to diagnose depression in streamed data. However, for the weak points in [49], the number of instances in this dataset is not large enough to

generalize the whole population. Therefore, sufficient instances are required to infer the results to the whole population. The approach used in this data collection [49] process can set a base for further academic research purposes.

Describing about another dataset [67], the dataset is made of tweets crawled from twitter enclosing the "#HCR" (health care reform) in order to validate the sentiments of the users. The tweets were manually labelled into 5 categories, namely negative, positive, neutral, unsure and irrelevant [67]. The dataset is further split into training, testing and development phases dataset. The main strength of this dataset is its methodology to validate the sentiment of the user, by manual annotation so that labels can be processed accurately. The main weakness of this dataset [67], is the number of tweets extracted from twitter is not enough for conducting research in predicting the depression. Although the tweets indicate to the health care domain, but not all of them are related to the mental health care.

3.2 Dataset Analysis - Comparison of Existing Datasets

This section outlines the comparative analysis of the available datasets covering various aspects based on the criteria including : Methodology, Used for Data Collection, Type of Features, Source of Dataset, Preprocessing, Methodology, Size of the dataset, Strength of Dataset, as well as Weakness of Dataset. Nevertheless, the tabular formation of previous dataset analysis helps to depicts various points to be kept in mind before constructing a new dataset from scratch or utilizing an old available dataset, which works on studying various factors such as selection of appropriate dataset to cover the objective of the research, knowing the pragmatic applications of the outcomes from the statistics performed previously on existing datasets, covering the strong points from all the previous datasets integrated together to build a new one, as well as finding methods to improve weaknesses from the previous datasets.

The table 2 represents the overview of comparison between all available datasets related to this study. The dataset [34] used scraping and crawling to obtain the data instances from Twitter. The large size of dataset and precisely defined features are the key aspects of [34]. However, the emotions are limited to just positive and negative. The dataset [37] used crowdsourcing to collect data and used association scores to indicate the emotion. The key aspect of [37] is the diversity in unigrams. However, it lacks the words which have a neutral impact. The dataset[38] collected dataset based on the keyword matching " I was diagnosed with". The dataset Includes keywords of multiple mental health disorders and applied exclusive conditions to build a highly precise dataset of affected users. On the other hand, dataset [40] collected data from user responses in order to assess the personality of the user. The dataset [40] followed approach proposed by Raymond Cattell based on the statistical

study. The key aspect of [40] is the validation of the genuineness of dataset by comparison of its method of extraction with the AMT's extraction methodology. Further, the dataset [42] contains responses of users from the questionnaire regarding three traits: depression, anxiety and stress on a 4-point scale. However, the size of the dataset [42] is less than the dataset [40]. Also, the self-rating of the user may not be able to diagnose mental health problems accurately. The Dataset [44] is the largest of all datasets in table 2 with 300 million Twitter users divided into three categories, depression affected, non-depressed and candidates for depression. The key aspect of [44] is the inclusion of visual features to explore the profile of the user in detail. However, the dataset is available at open-source level, therefore risking the privacy concerns of the users. Nevertheless, the dataset [45] Gathered data through teleconferencing and face-to-face clinical interviews. The use of humans as well as human-controlled agents to precisely diagnose depression helped to validate the responses. However, the dataset does not involve the user behavior on social networking sites. The dataset [46] worked on the concept of using actigraph as human-computer interaction technology to monitor the repercussions of depression. However, the size of the dataset is very small containing only 56 records. On the other hand, the dataset [49] scraped Twitter Data with 10k of user tweets containing necessary parameters to detect depression affected users. However, the size of the dataset [49] is less than the size of dataset [34],[38] and [44]. Lastly, the dataset [67] scraped data using Twitter API with "#HCR" . It contains tweet id, user information, tweet, sentiment, and comment. The key aspect of [67] is the manual annotation of labels which help to validate the outcomes from the classifier under the preprocessing methodology. However, the size and features of dataset [67] are less as compared to dataset [49],[44],[38] and [34].

Table 2 Showing comparison between different Datasets Available

Dataset	Data Collection Methodology	Dataset Feature	Dataset Source	Preprocessing Methodology	Dataset Size
[34]	Scraping	6 attributes and polarity of the data instance.	Twitter	Natural Language Processing (NLP).	1.4 million tweets
[37]	Human-based computation	14k unigram words and 25k senses	Crowdsourcing	Association scores to indicate the emotion	Dictionary of approximately 25k words
[38]	Keyword Matching	Diagnosed users and control users.	Reddit	Exclusion conditions	9000 affected users and 107000 control users
[40]	User responses	Personality traits.	16PF Questionnaire	Factor analysis	49k users with 166 responses of each user.
[42]	User responses	Depression, anxiety and stress	DASS Questionnaire	Users rated traits on a 4-point scale	39k users with 75 responses of each user
[44]	Twitter API extraction	Textual and visual	Twitter	Categorical divisions of users	300 million twitter users
[45]	Clinical interviews	Acoustic, video, sound and health-related parameters like ECG record.	Face-to-face and Teleconferencing Interviews	Manual annotation of labels	Includes 189 sessions of length 7 to 33 minutes
[47]	Actigraph wearable device	Monitoring patient's daily activities	Users activities	Mathematical variables to analyze Actigraph data	23 Schizophrenia And 23 currently depressed patients
[49]	Data scraping	40 Attributes of Tweet	Twitter	Extensive data cleaning	10k tweets
[67]	Data scraping	Tweet id, user information, tweet, sentiment, and comments	Twitter	Manual annotation of labels	2500 tweets

3.3 Data Collection - Building a novel Dataset for this Research

To begin with, we divide the research into 2 modules where the first module holds the experimental part of the research whereas the second module holds the proposed solution. Further, this division allows us to work on a 2-way architecture where we can work on a Tweet to Tweet level and a User to User level architecture under Experiments as well as the proposed solution respectively. However, this concept involves collecting 2 different datasets in such a way that the user and the tweets are connected either on a time series of tweets by each user or either by a similar sentiment attached with a unique id assigned by twitter API that was used to extract a particular tweet. Moreover, for validation of the prediction results to higher accuracy and recall value output, the Tweets under the depressed class for both the Experimental and the User Level module were taken from those users who have publicly self-declared for being diagnosed with depression in their Tweets.

To add to it, the collection of datasets for both the User Level and the Tweet Level were done from Twitter using tweepy library and twint tool. The Tweet Level dataset consists of unique tweets from 119240 unique users, where 29997 unique tweets are labelled as Depressed Tweets, taken from users who have publicly self-declared for being diagnosed with depression in their posts and posted tweets containing keywords related to the symptoms of depression (for example depressed, suicide, lonely, etc.) and the remaining 89243 Tweets are labelled as Non-Depressed Tweets containing keywords with highly positive sentiment (for example happy, enjoy, fun, adventure etc.) However, the User Level dataset consists of tweets posted by 760 unique users for a period of one month, where the tweets for 322 users out of 760 users were extracted according to the month under which

the user has self-declared for being diagnosed with depression and the tweets for the remaining 438 users were taken according to the month where they posted tweets containing keywords with highly positive sentiment (for example happiest, enjoying, amazing). Further, the tweets from 322 users were labelled as depressed class and the tweets from 438 users were labelled as non-depressed class for higher classification under the User Level. To explain in brief, for the experimentation module we collected a total of 1,39,346 Tweets which splits into 1,00,496 Tweets belong to Non-Depressed Category and the remaining 38,850 falls under Depressed category. After cleaning the tweets, the output for the number of tweets to be used for the experiments were modified because some of the users were showing duplicate entries, some of the tweets were holding a null value which means the information from the filters cleans out the keywords or meta tags from the tweet, as well as some of the resulting tweets were no longer involving the desired output. To epitomize, if a person has posted a tweet that holds a URL such as "<H1>: http:// Iamfeelingdepressed.alt </H1>" gets collected due to the string or keyword matching approach of an input given to Twitter intelligence tool known as Twint [65] and if the filter in the cleaning process holds a command to remove the posts with URL, then the model will ultimately end up removing that tweet from data to be analyzed. This process involves various data cleaning methods used in this research which will be discussed further in experiments and proposed solution methods. Nevertheless, after cleaning and processing the information collected, the final dataset for experiments holds 1,20,406 tweets (1 different tweet by each unique user) out of which 29,997 unique depressed tweets posted exclusively by its own unique user as well as 90,409 unique non-depressed tweets posted by their own disparate user. To add to it, the data then again passes through a final data

cleaning process which parses each tweet word by word and removes the unwanted items conclusively deleting 1,166 records which makes the dataset for experimentation module to be holding 1,19,240 unique tweets, each posted by a unique user as shown in Table 3.

Table 3 Collection of tweets for Tweet Level Architecture.

Type	Data Collected	Data cleaning	Pre-Processed Extensive Data Cleaning
Depressed Tweets	38850	29997	**29997**
Non-Depressed Tweets	100496	90409	**89243**
Total Tweets	139346	120406	**119240**

Table 3 Shows the collection, Data cleaning as well as the Extensive Data cleaning of Tweets used for the Tweet Level Architecture. The extensive data cleaning help to remove the copies of Tweets or users under both depressed and non-depressed classes to avoid biased outcomes and produce unique records in both classes.

Table 4 Final Dataset used for Tweet Level Architecture

Type	Experimental Dataset
Unique Depressed Tweets	**29997**
Unique Non-Depressed Tweets	**89243**
Total Tweets	**119240**

The table 4 shows the final Dataset used at Tweet Level Architecture.

3.4 Dataset for Proposed Solution

Moving further to the other side of this research, which is the proposed solution, the data collection originates from the users integrated onto the tweets analyzed by performing random sampling on the experimental dataset. This helps us to include the users who have a piece of higher relevant information regarding the category of the depressed or non – depressed sentiment conveyed by their tweets.

To instantiate, if a user X has posted “It was a depressive movie.” and another user Y has posted “I am taking anti-depression medicines.”, then ultimately the user Y will hold a higher depressive sentiment value observable by performing semantic and lexically analysis of each word from the statement showing that the user Y is already in depression and taking the medication whereas in case of user X the sentiment is weaker because the sentiment is not permanent as well as does not indicate towards a condition of user X having depression.

To conclude, a list of 760 users with highest intensity of sentiments was collected from the total users taken within the experimental dataset, under which 438 users were taken for a Non- Depressed User category and 322 users were taken for a Depressed User category as explained in section 3.3. However, the technique used to collect the tweets for each user works by extracting the tweets from the month in which the detected depressive tweet was posted to all the tweets posted till the month of February 2020. However, for the implementation of proposed solution, the tweets were sorted on a basis of a rule where a total of all the tweets of each user were taken only from the whole month of which the depressed or non-depressed tweet was detected.

To add to it, this data collection stores a total of 6,72,386 tweets from 760 users, out of which 3,32,352 tweets are posted by 438 Non-depressed users and the remaining 3,40,034 tweets are posted by 322 Depressed users. Nevertheless, after performing extensive measures used to clean data, the resulting dataset maintains a number of 2,99,539 tweets from the total collection of 6,72,386 records, out of which 2,12,909 tweets posted by 438 Non-Depressed users whereas 86,630 tweets created by 322 Depressed users are collected to build the final dataset for Proposed solution. The outcomes are shown under table 5 and table 6.

Table 5 Collection of tweets for User-level Architecture.

Type	Data Collected	Pre-Processed Extensive Data Cleaning
Tweets of **322** Depressed Users	3,40,034	**86,630**
Tweets of **438** Non-Depressed Users	3,32,352	**2,12,909**
Total Tweets of **760** Users	6,72,386	**2,99,539**

Table 5 Shows the Collection of Tweets and Data cleaning of records for user level architecture.

Table 6 Final Dataset used for the User Level module.

Type	Proposed Solution Dataset
Tweets of **322** Depressed Users	**86,630**
Tweets of **438** Non-Depressed Users	**2,12,909**
Total Tweets of **760** Users	**2,99,539**

Table 6 Shows the records for the final dataset generated after data cleaning of the records used at user Level Architecture

3.5 Comparing the Dataset for Tweet Level and User Level

The dataset for the experimental module was used to predict the depression in users on a Tweet Level. However, the dataset for the proposed solution was used for an in-depth analysis as well as understanding the pattern of the users on Twitter. Both the datasets are different from each other in terms of the following factors:

1. The Tweet Level dataset contains unique tweets posted by 119240 unique users, whereas the User Level dataset contains multiple Tweets from each of the 760 users for a period of one month.

2. The attributes in table 11(a) and 11(b) shows all the components of a Tweet with the user's profile that are used for visualization under the Tweet Level. However, only the extracted text of the Tweets was used for further implementation in the experimental module. On the other hand, the user level dataset contains the components of Tweets with additional 12 attributes of a user's profile in terms of Feature Engineering Criteria (section 6.2 (E)) operated under the proposed solution.

Chapter 4: Gap Analysis, Research Questions, And Problem Statement

This chapter covers a gap analysis carried out using previous research papers and datasets to focus on an objective that aims to fill the voids raised from the weaknesses as well as the missing features from the past studies. Further, research questions stemming from various aspects are portrayed. Lastly, a discussion for the challenges to achieve the goal of this research are reported in the problem statements.

4.1 Gap Analysis

To begin with, a structure where a lot of researchers focused on the prediction of depression level of an individual or depression amongst users by using data available on social media. Various researchers target to form datasets using posts created by users on social media like twitter and other web blogs for data extraction by tracing symptoms of mental illness in the context of content available publicly whereas some researchers followed data collection approach by other methods such as Web scraping tools, online surveys, and questionnaires.

Out of all the resources available in this modern world for discovering so much from statistics and results from existing research papers, there are still some unrevealed details hidden in some concepts of extraction of data, processing the datasets, location-based collection of data, methodologies as well as various other techniques used in different research. To find those weaknesses of the existing research integrated together for a better understanding, this gap analysis was conducted to define a clear picture of the ongoing trend. Moving further towards detailed discussion some researchers have found good results, but they lack to measure different types of changes in behaviour of depressed

and non-depressed users on social media such as night time engagement, pattern in the selection of words under various topics for depressed and non-depressed users using Topic modeling concepts etc.

On the other hand, another research paper moves forward with a different approach of building up a database using crowdsourcing method focused on twitter dataset collection that comes with the limitations of twitter users which are private or shared their posts within friends only and users in depression with no twitter accounts, as well as users which share the posts within other platforms such as Facebook, Instagram, and other web blogs gets unmonitored.

To add to it, Language also plays a vital role in the processing of datasets because the posts which might be containing traces of depression symptoms are eliminated by the models due to the language conversion barrier manipulating the accuracy of the result. Furthermore, some researches have unorganized structures to represent the statistics of dataset studied visually and graphically to show an understandable comparison between various features used as well as factors influencing each model.

Besides this, a lot of research has been conducted based upon keyword matching, Textual information and already readable posted data sources whereas there are some sources which are not in the form of directly readable content such as images, video as well as audio sources that should be dealt as targets for future work.

However, there has been some research-based upon audio feedback and image processing, but it does not prove higher accuracy in context with the analysis of predicting depressive symptoms. Therefore we could conclude that the text analysis could be combined with a model that could enhance the transcripts from audio feedback as well as

an image processing unit which could process the images to read the text by Optical character recognition and the texts generated from videos or any social media by each user to develop a more accurate model for further investigation to help detecting depression. On the other hand, it will be challenging to breakdown a video into image readable frames and then processing with addition to a database that integrates audio converted into text transcripts and the posts generated by the user.

However, this concept seemed to be a part for the future work where such a model requires high GPU and CPU computational performance hardware as well as statistical mathematical calculations needed to convert those frameworks and the information from the audio, video as well as the textual content into a computer-readable form of data and finally into a CSV dataset file for further analysis.

Moving further, some research outputs lack pharmaceutical keywords to match the posts which are related to medicines that affect the mental health of the mother during pregnancy [5]. To add to it, one of the research papers limited the evaluation criteria by using only five types of emotions which snag to get higher accuracy because of the other factors that could be considered such as post time, the relationship between the content posted before and after the post that reveals symptoms of depression.

Nevertheless, some research papers follow a common gap analysis in which they lack to use a control dataset that should not contain any keywords in the context of symptoms related to depression in order to avert unfairness to outcomes inclined towards a biased result accuracy for the dataset with depressed records. To add to it, the comments could also be used in a prolific manner by inspecting them further into linguistic features to find more useful posts relating symptoms with mental illness. Sometimes the vocabulary

becomes a major issue in detection of the textual analysis for the exact keyword matching that has been used in a post which reveals an attempt of suicide by a user. Such posts can be targeted to make a pragmatic model promote the identification of the user before any pernicious event takes place in the future. Among other things, there exists a hitch where the location is a big concern because the collection of data should not be made specifically to extract information from a fixed particular area, city, province and country. To add to it, there exist many situations under which one part of the area targeted may not be suffering from similar types of issues regarding mental illness as compared to the locations which are unmonitored leading towards a false resulting outcome.

On top of that, the technical part should be made more simplified to understand which relates to the breakdown of each and every part of the posts that gets undiscovered. For instance, the hashtags, emoticons, user mention, and other demographic attributes can be further analyzed for a better accurate prediction of depressive symptoms. Above and beyond, some research papers have used only some specific models such as SVM, Random forest as well as logistic regression which gets skimpy when it comes to a comparison with other models such as ANN and LSTM. Also, there exists another problem with the comparing of two research papers when both are using different datasets but the same models or vice versa.

4.2 Research Questions

To begin with, the background study helped to understand a lot of work that has contributed towards the technologies as well as methods used previously in order to predict depression using different machine learning techniques. On the contrary, there exist certain

anomalies that allow a lot of future work that could be done in order to revamp the whole concept of investigating depression more meticulously by targeting the focus on unobtrusive points which are discussed in the gap analysis section. Moving further towards this research which will help to find a germane solution, a lot of questions emanates in context to developing a novel machine learning model and its working architecture to predict depression in users on social media. Into the bargain, this research will help divulge some of these research questions stated further as follows:

Q1. How can we make a machine learning based model that can automatically process and predict depression from different tweets?

Q2. What is the relation between different types of attributes involved in the dataset constructed using the real-time extraction of tweets?

Q3. What type of characteristics we can extract from tweets to classify users into the categories of depressed and non-depressed user ?

Q4. What kinds of analytical facts can be recognized for users in different age groups on social media platform (Twitter)?

Q5. What other demographic attributes influence the detection of depression on social media?

Q6. What types of patterns in vocabulary and sentence construction can be observed from different types of users on social media.?

Q7. How can we utilize the features such as location and user engagement data available on twitter?

Q8. Which machine learning techniques work the best to detect depression in a user from their tweets on social media and how can we validate the accuracy of such prediction?

4.3 Problem Statement

First and foremost, so many tests, surveys, and questionnaires arise to investigate the detection of depression in a person but holds a lot of drawbacks at the same time as well. However, in the long run, it's very complex to cover large masses of people around the world with these methods. Moreover, converting inputs from these methods having distinctive factors, to interpret the same emotions and feelings as the output can be altered by factors such as location, language barriers, privacy concerns, incomplete or ambiguous information. Consequently, data science researchers start to target depression among social media users where social media platforms like Facebook, Twitter, Instagram, and various web blogs play a vital role in covering a large population as well as holds a reach of endless users connected around the world digitally.To add to it, Studying the gap analysis from previous research papers gives a layout for the missing concepts that could be applied for creating a better model that is able to predict depression in users using machine learning. However, these concepts for predicting depression using social media create a lot of problems that are discussed further in this section.

To begin with, Twitter was selected as an appropriate source of data because of it’s real time and publicly available content for extraction. However, the drawback with Twitter data extraction was the maximum limit of 3200 tweets per user which blocks the extraction request once 3200 tweets per user gets fetched. Also, the manual collection of tweets for each user consumes a lot of time for building a larger dataset. Moreover, there may be many instances where the same user has created different tweets with same keywords which might cause a duplicate information while extracting Tweets for experiments at Tweet Level.

Moving further, most of the researches either worked on the tweet level or the User level where this research comprises of both the two architectures such as the Tweet Level under experiments and the User Level under proposed solution. The basic idea for building both the architectures were interlinked by using the same users that were extracted under experiments. However, the problem arises while selecting the number of appropriate users because the unique tweets posted by the users may be more than one which will repeat the same user while selecting a list for users under the proposed solution. Besides this, after the data collection there exists another problem while cleaning the data where lot of previous researches loss an essential part of the Tweet related to emotions from the emoticons. Apart from this, understanding the importance of accurately detecting the results for each prediction is an essential part, which makes it a difficult choice while selecting an appropriate classifier that will produce a high recall value. In addition to it, ensemble methods such as AdaBoost, XGBoost, Gradient Boost and artificial recurrent neural network such as LSTM and BI-LSTM could also be implemented other than SVM model used commonly in most of the previous research works. Extensive Feature Engineering can be applied to increase the models performance, because most of the previous researches relies on textual information and does not involve other features that provide more information for understanding the pattern of user on social media such as age, location, social response ratio, lexical features of vocabulary, polarity contrast and Topic modeling. On the other hand, selecting the appropriate range of time and number of unique tweets by each user can be problematic for data collection because the timeline for informative tweets might be differ for each user. Aside from this, some of other problem statements are as following:

1. Understanding the Frequency distribution of words and utility of own trained word embeddings into vectors used under Word2Vec with TF-IDF.
2. Creation of Age variable to reveal factors related with engagement of user on social media belonging to different age groups.
3. Data Collection of tweets from each individual user profile for the whole month under which the indicator of depression has been detected.
4. Integrating the application of Twint - Twitter Intelligence Tool and Tweepy python Library in formation of a robust method to build datasets at tweet level and user level.
5. Data Cleaning of ambiguous and duplicate information as well as attaining demographics from unique users with respect to their unique tweets.
6. Implementation of the concepts of (Part of Speech) POS tagging using Natural Language Toolkit (NLTK).

4.4 Challenges

To solve the questions arising from the gap analysis and problem statements a lot of challenging problems faced are discussed in this section. Further, of all the social media platforms Twitter is considered as the most popular to fetch information among data scientists because of its publicly open-source nature and ease of scanning user data within the small size of tweets with only 280 characters. On the other hand, it comes with a lot of drawbacks such as while utilizing a machine learning tool or already existing python libraries for extraction such as Tweepy [66], there exists a gap when the Twitter API blocks the process at reaching maximum of 3200 tweets for each user which gives a lot of problems when deriving a large time series of tweets for each user on a larger set of data. Also, taking more time and manual effort to re-run the command again and again. Further,

Tweepy [66] has a weaker link as compare to Twint [65] when it comes to keyword matching and loading bag of words approach within the use of a rigorous dictionary to fetch the exact and specific information which results in a lot of wrong and unwanted tweets resulting in difficulty for cleaning and preprocessing producing less accurate results.

Moreover, Twitter API allows every information given by a user publicly but does not allow to extract the age as well as the date of birth of the user which makes it difficult to understand the concepts behind content and statistics related to age demographics. Moving further, Twint [65] works very precisely and with an explicit approach without any twitter account or access token permissions also it does not block when reaching the limit to stop and also works with a set of a time series frame where the start and end dates can be used to fetch information during a particular timeline.

On the other hand, when it comes to pulling the attributes such as followers count, retweets, the following count, favorites as well as the geographical location for a given list of a large number of users, Twint [65] works with only one user at a time and requires manual re-run of the command again with the terminal for every input. However, Tweepy [66] works smoothly with a large number of users list and their attributes to be collected together on a single loop. Apart from this, real-time data cleaning is a very complex process where in most cases researchers remove emojis and hashtags which could be useful if converted into a meaningful form using Emojipedia API connecting with a python emoji library to detect emojis which could be further visualized to see statistics based upon most emojis and type of emojis used by depressed and non-depressed users as well as relationship between the emojis, hashtags and the sentiment of the tweets.

What is more, a lot of research papers lack feature engineering despite gaining a higher accuracy but holds a less precision and recall value due to incomplete use of sentence and grammar connected with adverbs as well as adjectives that could be converted into a hybrid dictionary holding lexical analysis of each tweet helping the model to train better resulting output. Moreover, there exists a lot of cases where the ratio for the train to test data exploits scores of outcoming results.

To instantiate, if the amount of data extracted is divided more among the count for training with fewer users then the remaining data for testing goes wrong with an overfitting problem giving a 100 % accuracy and precision value of 1. However, as a trend seen under some research papers where nearly 100 depressed and 300 non-depressed users were taken with a tweet count of 1 year of their backtracking data. This approach may give a biased result because the timeline for informative tweets might be different for each user. Also, most of the tweets related to symptoms indicating a depressed sentiment are done between a 1-2 months of the tweet indicating self-declaration of having depression, and the remaining tweets might be taken as a positive sentiment which increases the total sample data for the Non-depressed class creating a under sampling data resulting in a biased output.

Further, this research has followed all these problem statements to understand the methods that help to overcome the issues seen from the past research papers and from a lot of samples of experiments where the amount of user dataset and other factors such as a ratio of 70 % test and 30 % train, or 80 % test to 20 % train as well as a sample of 60 % train to 20 % test with a validation dataset of 20 % were considered and the best working sample was selected for the experiments for this research discussed in the next sections

Lastly, the goal of this research is to develop a novel technique or a machine learning model with a motive to detect and predict depression in users on the social media platform. The model will help to detect a depressed user even if the self-declaration of having depression is not posted publicly by the user. The model utilizes user's previous and current posted tweets, pattern of engagement on social media (Twitter), behavioral sentiment of the tweets, as well as unique concepts applied under feature engineering to predict the depression in user. Further, the confusion matrix helps to inspect the test sample with an accuracy, precision and recall value of the results obtained. Finally, the results of k- fold cross validation score are obtained to ratify the average accuracy with the working of each model.

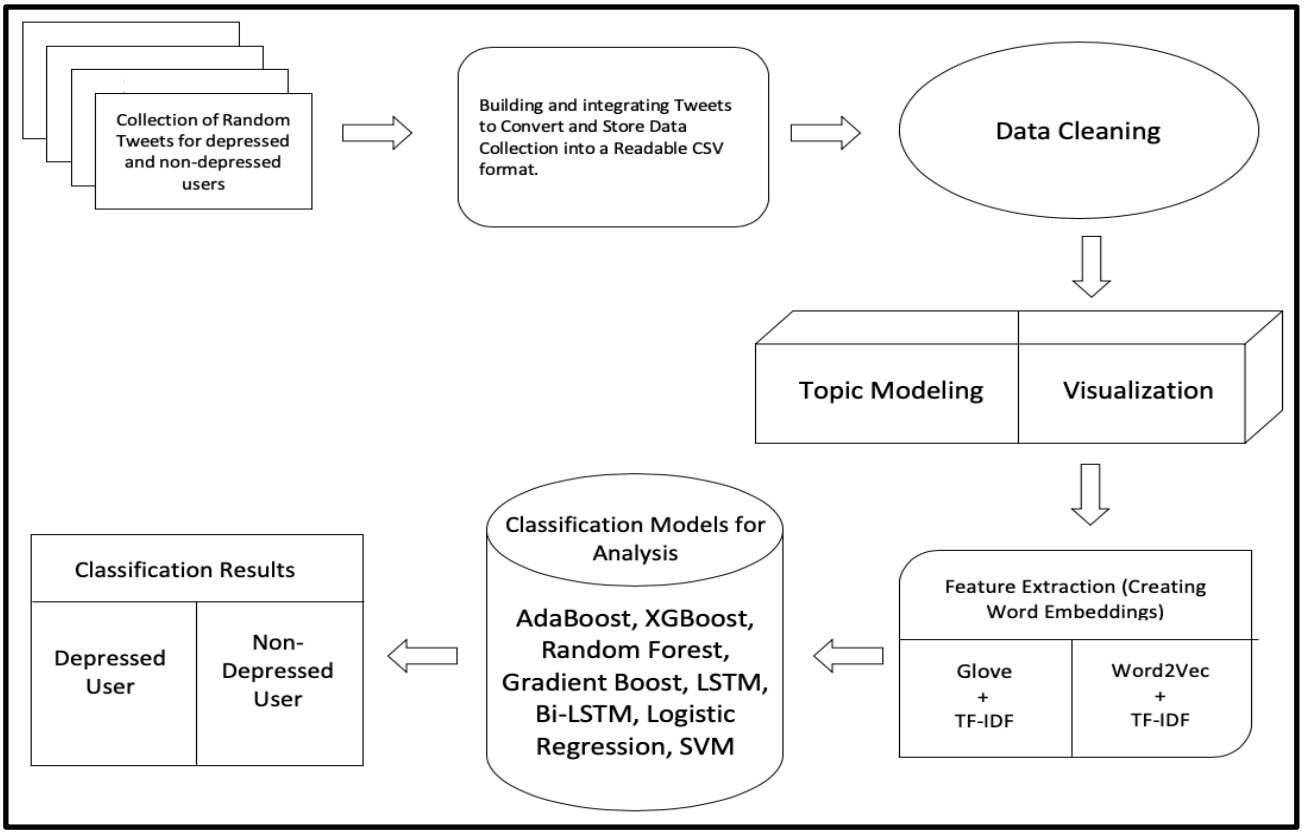
Chapter 5: Experiments

This chapter covers the methodology and technical aspects of this research followed under the tweet level architecture, which helps to understand the steps behind the tweet level investigation for detecting depression. Further, an explanation of each process including data collection, data cleaning, topic modeling, feature extraction, word embeddings, LDA, as well as advanced machine and deep learning algorithms by using ensemble learning methods are discussed. Lastly, the representation of the dataset in an analytical form is carried out under visualizations.

5.1 Tweet to Tweet Level Architecture

The procedure to investigate a tweet under a depressed or a non-depressed category falls under a technique of natural language processing (NLP) which helps a model to process the information similar to a behavior of human brain to convert the unstructured sentence into a meaningful form with a sentiment, lexical as well as semantic analysis performed on the background that allows the model to declare the opinion of the input context. Into the bargain, a system architecture was created for following the operating pattern of this research shown in Illustration (1).

Illustration 1 System Architecture for Workflow of Experiment



Furthermore, this system architecture was created as a roadmap to understand and highlight the concepts followed behind the process on each stage. The first stage involves the data collection for which a twitter API scraping tool was required which could help to load a maximum number of tweets within a unique and rigorous use of phrases to be considered onto a lexical and semantic scale of the tweet to be considered for extraction, running in a parallel process.

There are various web scrapping tools available which works with the twitter API out of which Twint [65] and Tweepy [66] are considered for this research because of their better performance and compatibility with machine learning models, where Twint [66] was operated for extracting the unique tweets posted by each unique user for the experimental

dataset analysis under tweet to tweet level architecture and tweepy [66] was used to pull each users profile details.

Withal, Tweepy [66] works with a python library to connect the twitter API and extract tweets as well as demographics for each user's public detail but due to twitter's privacy concern of allowing a limit of 3200 tweets per user extraction data, we use Twint [65] for building the proposed solution dataset which does not require any access tokens and works without a Twitter API. Twint [65] runs as a machine learning technique proficient to maintain a continued flow of the tweets with an inbuilt lock function that manipulates the block of status error from twitter and allows to fetch more than 3200 tweets for each user. However, the collection of each user's profile demographics was mined by using Tweepy [66] and integrated together with the proposed solution dataset under user to user level architecture for visualization and other analysis discussed later in the proposed solution. Nevertheless, to run a latest version Twint [65] v2.1.14, the command terminal needs to initialize the installing function "pip3 - install - twint" and setup the environment as per the requirements of data to be extracted by installing some of the basic libraries which includes Geopy, python Version 3.6, beautifulsoup4, Pandas [65] etc.

5.2 Data Collection

The experimental dataset collection process explained previously in the chapter 3rd is followed at this stage, which represents the collection of tweets carried out by passing the statements in Twint [65] , also defined as a Twitter intelligence tool to fetch a distinct tweet created by a user holding a unique User ID. Further, the phrases used to pull the tweets by

using Twint [65] for Depressive as well as Non-depressive categories on terminal is shown in Illustration 2

Illustration 2 showing Twint command on terminal

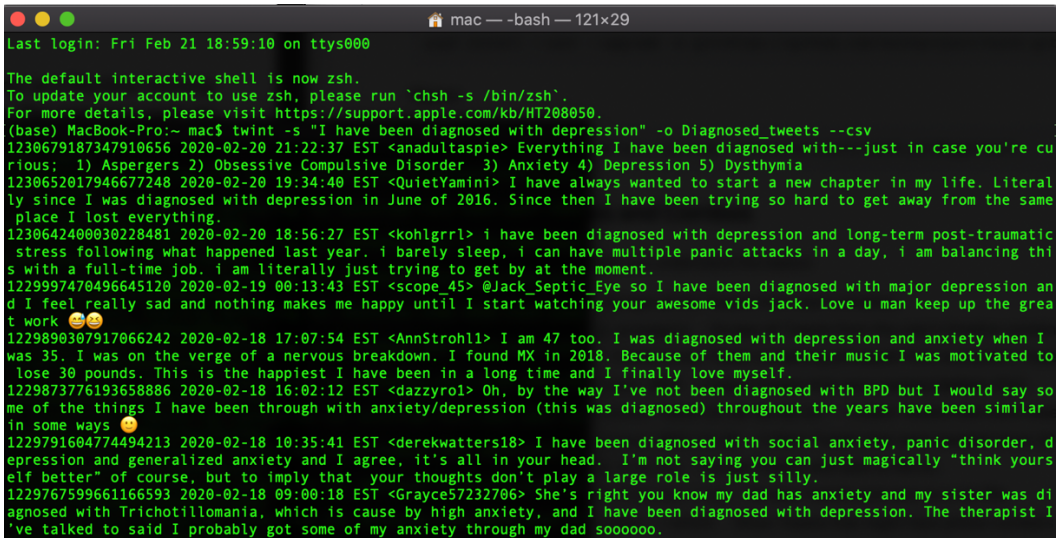


Table 7 Showing phrases used for Non-Depressed Users

"I am joyful"
"I am amazingly happy enjoy"
"I am enjoying happy"
"I am enjoying travelling
adventurous"
"I am enjoying travelling "
"I am feeling amazing "

The table 7 and table 8 shows the phrases used under the depressive and non-depressive category to be passed in the Twint[65] command. However, for construction of a high-quality dataset, tweets related with the matching of some specific sentences were utilized

as a key aspect, which helps to avoid the unwanted or false inputs for better data cleaning and further pre-processing.

Table 8 Showing phrases used For Depressed Users

"I attempted suicide depressed"
"I am anxiety patient"
"I am suffering from anxiety"
"I am feeling suicidal thoughts"
"I Blurred no idea of life stressed"
"I have been diagnosed with depressed"
"I have been diagnosed with depression"
"I am on depression medications"
"I am suffering from depression"
"I am poor mental health"
"I sleepless nights depression"
"I irritate depressed alone"
"I attack nausea episodes"
"I overweight depression"
"I am under treatment depressed"
"I am on antidepressants"
"I take drugs depressed"
"I am under chemotherapy depressed"
"I have been diagnosed with Postpartum depression"
"I am suffering from postpartum depression"

To add to it, for the profile details of each user taken under tweet to tweet level, Tweepy [66] was used by importing libraries shown in table 9 as well as generating active consumer secret keys and access tokens as shown in table 10.

Table 9 Libraries imported for Tweepy

```
import pandas as pd
import tweepy
from tweepy import Cursor
import unicodecsv
from unidecode import unidecode
```

Table 10 Authorization of twitter API

```
consumer_key_twitter =
consumer_secret_twitter =
access_token_twitter =
access_token_secret_twitter = "
authorization_access = tweepy.OAuthHandler(consumer_key_twitter,
consumer_secret_twitter)
authorization_access.set_access_token(access_token_twitter,
access_token_secret_twitter)
api = tweepy.API(authorization_access)
```

Further, the extracted raw data gets added into a csv file along with various features such as username, followers, following, location and other attributes related to the user are represented in the table as shown in Table 11 (a) and 11(b).

Table 11 (a) Showing attributes of twitter user

Name of the Feature	Datatype	Description	Sample of Value stored in feature
id	string	String value uniquely identifying the tweet	“id” : 1225450042913427456
Conversation_id	string	String value uniquely identifying conversation thread	"conversation_id": “1224454674238980097”
date	string	Date including year, month and day at which the tweet was posted	“date” : “2020-02-06”
Time	string	Time including hours, minutes, and seconds at which tweet was posted	“time” : “11:03:52”
Timezone	string	Describing the time zone of the user	“timezone” : “EST”
User_id	string	String value containing numbers that uniquely identifies the user	User_id : {‘168389885’}
Person_username	string	Unique username of the person to which the Twitter account belongs	Person_username : {‘spinnersdance’}
Person_name	string	Name of person to which the Twitter account belongs	Person_name : {‘TheLawOfLisa’}
Person_followers_count	Int 64	The number of persons followed by the person	Person_followers_count : {‘555’}
Person_following	Int 64	The number of people, the person is following	Person_following : {‘955’}
Person_favorites	Int 64	Count indicating the number of items the person has marked in favorites	Person_favorites : {‘4045’}
Person_verified	Boolean	Value depicting whether the profile of the person is verified or not. Contains value True or False	Person_verified : {‘True’}
Person_location	string	Location of the person as declared by the user in its profile	Person_location: {‘United States’}
Person_statuses_count	Int 64	Count indicating the number of status of the user	Person_statuses_count : {‘67’}
Person_description	string	Short sentence usually containing the brief description about the person	Person_description: {‘dance teacher Spinners Dance Studio inc Class Act Theatre School’}
Person_geo_enabled	Boolean	Value indicating whether the person has given access to Twitter to record thx location	Person_geo_enabled : {‘True’}
Account_creation_date	string	Indicating the date when the person has created the user account on Twitter	Account_creation_date: {‘2009-03-15 20:40:10’}

Table 11 (b) Showing attributes of twitter user

text	string	Containing the tweet which the user has posted on its timeline	Text : {‘Just a guess, but a little far from home if it is ’}
source	string	Value indicating the source device from which the person has tweeted	Source : {‘Android’}
Retweet_status	Boolean	Value indicating the status of the tweet whether it is retweeted or not	Retweet_status: {‘True’}
Retweet_count	Int 64	Value indicating, the number of times, the withe context was retweets	Retweet_count : {‘3’}
User_mentions	string	The names of the other persons, the user has mentioned in his tweet.	User_mentions: {‘bellletstalk’}
Hashtags	string	The hashtags that are written by the user in the tweet	Hashtags: {‘#depressed’}
Hashtags_count	Int 64	Value indicating the number of hashtags present in the tweet	Hashtags_count: {‘3’}
Urls_count	Int 64	The number of urls mentioned in the tweet	Urls_count : { “ 0 ” }
User_mentions_count	Int 64	The number of user mentions total present in the tweet	User_mentions_count : {“1”}

5.3 Experimental Setup

This section represents the discussion of technical aspects of this research including initializing the libraries in python, data collection, LDA, combinations of word embeddings, description of ensemble learning models with their pseudo codes as well as methods to calculate results used in this research.

5.3.1 Loading Libraries

To run the experimental module, the first step involves the importing of various python libraries that helps to perform various statistical and graphical analysis, plotting visuals as well as loading functions that work with training and testing of algorithms used in this research. Some of the important libraries include Tweepy, pandas, seaborn, nltk, numpy and keras, gensim, geopy, tensor flow, as well as sklearn.

Table 12 Showing pattern of important libraries imported

```
1. # pandas for data manipulation in rows and columns
   import pandas as pd
2. #importing stopwords from nltk
   from nltk.corpus import stopwords
3. #loading stop words
   stop_words = stopwords.words('english')
4. #numpy for matrix manipulation
   import numpy as np
5. #load TF-IDF vector
   from sklearn.feature_extraction.text import TF-IDFVectorizer
6. #import regular expression for text manipulation
   import re
7. # loading metrics for checking accuracy
   from sklearn.metrics import classification_report, accuracy_score
   from sklearn import metrics
8. #importing natural language toolkit for text processing
   from nltk.stem.snowball import SnowballStemmer
   import nltk
9. #import csv module of python
```

```
import csv
#importing word cloud for visualization
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
from pprint import pprint
#library to train word2vec embeddings
import gensim
#library for visualization
import seaborn as sns
#importing components for statistical classifiers
from sklearn.model_selection import train_test_split
from sklearn.metrics import plot_confusion_matrix
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error
from sklearn.metrics import mean_absolute_error
from sklearn.linear_model import LogisticRegression
#preparing dictionary model for topic modeling
import gensim.corpora as corpora
from gensim.utils import simple_preprocess
from gensim.models import CoherenceModel
#importing math module for basic calculations
import math
#importing modules for lstm and bilstm
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import LSTM
from keras.preprocessing.sequence import pad_sequences
from keras.preprocessing.text import Tokenizer
from keras.utils.vis_utils import model_to_dot
# for handling the warnings
import itertools
import warnings
#for the application k-fold cross validation
from sklearn.model_selection import cross_val_score
#metric for checking precision
from sklearn.metrics import precision_score
#importing xgboost classifier
import xgboost as xgb
#function for checking the time utilized by algorithms
import time
#importing matplotlib for visualization
import matplotlib.pyplot as plt
```

5.3.2 Loading and combining depressed and non-depressed Tweets

First the depressed as well as non-depressed tweets were loaded for checking of any duplicate instances. After removal of same users or same tweets, the resulting tweets were arranged and combined into a single input of data frame which contains unique tweets with attributes from each tweet as shown in Table 13. Nevertheless, the target variable is set as '0' for non-depressed tweet and '1' for depressed tweet.

Table 13 Showing attributes of each tweet

Index (['id', 'conversation_id', 'created_at', 'date', 'time', 'timezone', 'user_id', 'username', 'name', 'place', 'tweet', 'mentions', 'urls', 'photos', 'replies_count', 'retweets_count', 'likes_count', 'hashtags', 'cashtags', 'link', 'retweet', 'quote_url', 'video', 'near', 'geo', 'source', 'user_rt_id', 'user_rt', 'retweet_id', 'reply_to', 'retweet_date', 'translate', 'trans_src', 'trans_dest'], dtype='object')

5.3.3 Data Cleaning and Preprocessing

Data cleaning is crucial task when preparing data for its processing in machine learning algorithms. Data captured in real-world scenarios might contain nonviable, noisy, irrelevant as well as illogical content values that can hinder the procedure for obtaining useful results. Data in incomplete and inconsistent format is also one of the reasons why data cleaning is pertinent before its processing. Additionally, data cleaning can generate small, clean subset of data, which can lead to high quality pattern analysis [68]. Procedures involved in data cleaning are detection of outliers, remove noisy data points, selection of relevant attributes, reducing number of rows, resolving data conflicts if any. Data cleaning ensures the avoidance of errors related to cost, increase productivity of the project as well as aids in better decision-making process. For cleaning the textual information present in the form of tweet's texts in the corpus, the data cleaning methodology involves loading

tweets into supported data structure (basically a list), and then removing unnecessary information in the tweets. The functions build for various processes in data cleaning are:

The data cleaning process initiates by traversing each document or tweet iteratively, character by character, and skipping all the irrelevant information, except the words that strongly contribute to the construction of the sentence. The removal of extraneous information includes the following procedures:

(i) **Removal of Punctuation Marks**: The punctuation marks are the symbols that support the construction of complex sentences. “String.Punctuation” is the functional module containing punctuation symbols that are matched with raw text. The frequent symbols of punctuation marks are “" ( ) * ^_`{|}~ , + , - . / : ; < = > ? , ! " # $ % & , @ [\ ]” that are when encountered in the sentence are removed from the sentence.

(ii) **Removal of HTML Tags**: Most of the real-world data scraped from online resources contains html tags such as <p>, <h1>,
, that are supplementing noise to the data and also during the information retrieval phase, can lead to more memory and space consumptions. This inoperable and the useless context is removed by using ‘BeautifulSoup’ library that facilitates the data cleaning process by extracting data from html and xml context-based files.

(iii) **Removal of Stop Words**: Stop words can be described as the most repetitive or common words occurring in the sentences. List of common stop words include “and, or, not, like, could, have, want, where, …” To filter out the relevant context in natural language, stop words are removed in the phase of data cleaning. This could significantly decrease the size of dataset, and perhaps the computation time, and also promotes better classification through machine learning models. Python’s Natural Language Toolkit has

the list of stop words of over 16 languages which is taken into consideration when removing it from corpora.

(iv) **Removal of URL:**The links related to posts and pictures that does not give insights to the sentiment of the user are removed using regular expression in NLTK.

(v) **Removal of Emojis and extra words**: The emojis from the text are removed in the tweets, and extra repetitive words present in the corpus, for instance, ‘sometimes, whom, maybe, com..’ were also removed from the tweets.

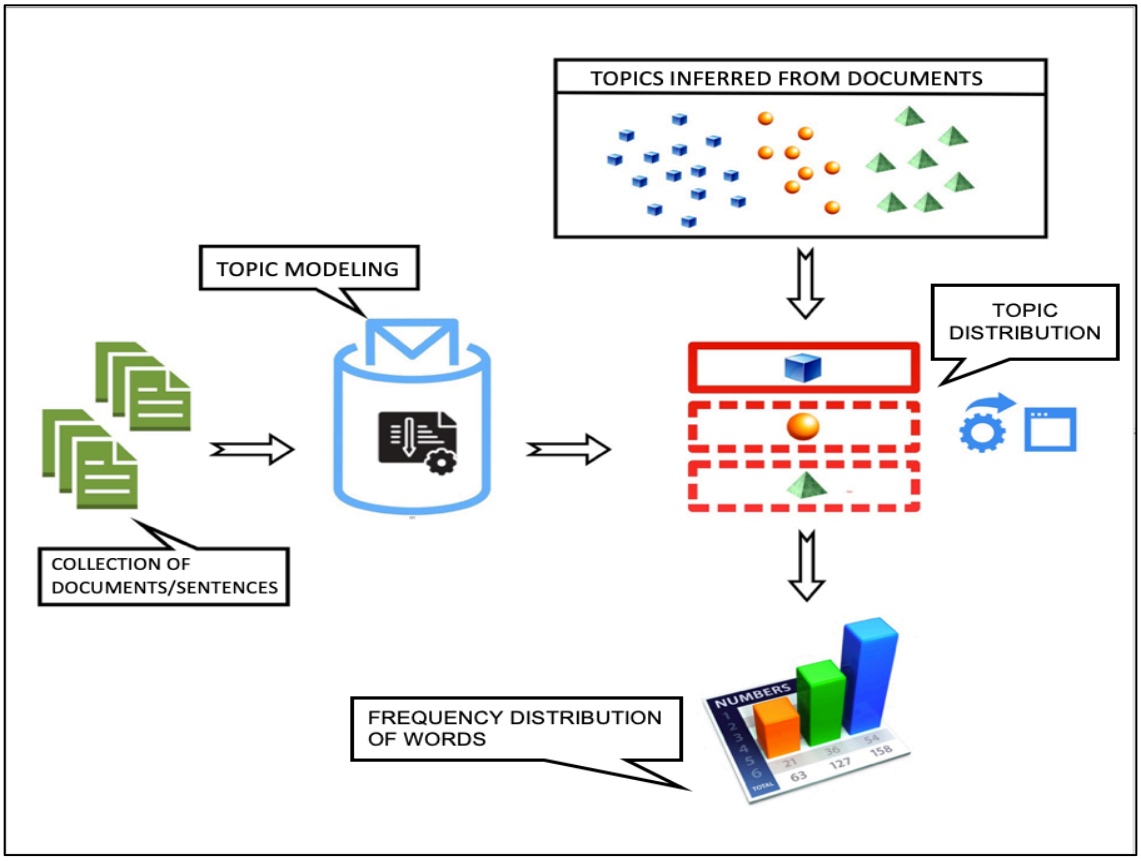
Table 14 Table for sentences presented before and after the procedure of data cleaning

Uncleaned Sentences (Depressed Class)	Cleaned Sentences (Non-depressed Class)
['4 years ago I attempted suicide, was extremely depressed, doing a lot of drugs. After a lot of hard work and patience, I am clean, I love myself, I am the happiest I’ve ever been. Never lose hope & put in the work ❤️#BellLetsTalk',]	['4 years ago attempted suicide extremely depressed lot drugs after lot hard work patience clean love happiest ever never lose hope put work bellletstalk',]
'When they say it gets better, they aren’t lying. This time 3 years ago I attempted suicide bc I was so depressed and hopeless. Flash forward to today and I have a whole ass husband and two babies. Yes, I still struggle, but I’m also happier and more loved than I’ve ever been.'	'when better arent lying time 3 years ago attempted suicide depressed hopeless flash forward today whole ass husband two babies yes still struggle im also happier loved ever',
'You can dm us both if need be. I had a "friend" years ago try to fake an attempted suicide and it turned out just like this. All signs point to it\'s Chris. He\'s depressed and he wants reassurance without directly asking for it. He needs mental help.'	'you us need friend years ago try fake attempted suicide turned all signs point chris hes depressed wants reassurance without directly asking needs mental help'

5.3.4 Topics Discussed in Depressed and Non-Depressed Classes

Real world data comprises of information in structured and unstructured format. There are many hidden factors laid out in data itself which gives detailed insights and contributes in analysis. Topic modeling using Latent Dirichlet Allocation (LDA) is one of the unsupervised methodologies used to find latent or hidden features present in the textual information. This technique empowers to analyze the topics inferred from the textual context in an unsupervised way, without defining the topic indexes to the documents before its application.

Illustration 3 Showing working concept of Topic Modeling Architecture



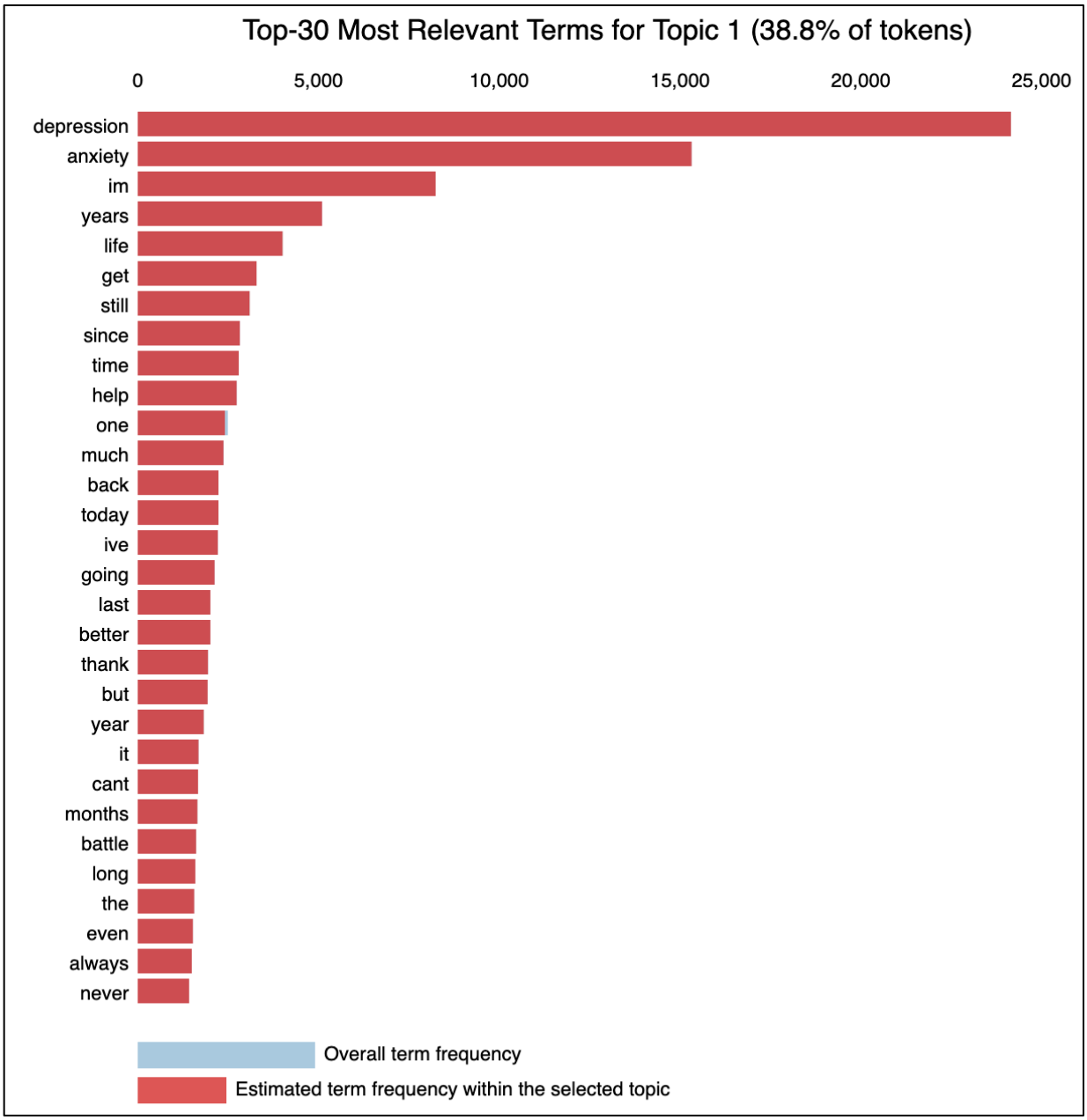
Latent Dirichlet Allocation (LDA) works on the assumption that the topics present within the corpus are generated from a mixture of topics. These topics further produce words depending on their probability distribution within the corpus. It is basically a technique incorporating the concept of matrix factorization. The collection of documents can be exemplified in Term-Document Matrix [88]. LDA processes this Term-Document Matrix to split into two matrices Topics-Document Matrix and Terms-Topic Matrix [88]. LDA further utilizes technique of sampling to refine the results of these matrices. For each iteration, it traverses through each term present in the document and maps the current word-topic label with a new label by calculating probabilities. For every single topic, two types of probabilities are premeditated, one describing the ratio of words that are assigned with the topic label in a single document and the other probability describing about the ratio of labels belonging to a specific topic across all sentences coming from a particular word. LDA considers two hyperparameters alpha and beta [88]. Alpha denotes topic-word intensity and beta denotes topic to word intensity. For topic modeling, Gensim library is used which takes number of topics and number of passes as the main parameter. To infer the topics, from this corpus, the values of hyperparameters and parameters are:

Table 15 Showing Parameters Passed in LDA

parameters	Depressed dataset	Non-depressed dataset
Number of topics	12	20
Number of passes	10	10

Moreover, Topic 1 for depressed data holds 38.8% of tokens and Topic 2 for depressed data holds 21.4% of tokens. Similarly, Topic 1 for Non-depressed holds 28.5% of tokens and Topic 2 for Non-depressed holds 14% of tokens.

Illustration 4 Showing Topic Modeling Results for Depressed data for Topic 1



The **illustration 4** depicts the dominance of words in topic one, when applied topic modeling to the depressed class. The results indicate that the top words in the topic one is depression, anxiety, better, life, always, never that indicate to the negative polarity. Similarly, the **illustration 5**, portrays the dominant terms in topic two of depressed class. The frequent terms are fighting, know, don't, really, love, etc.

Illustration 5 Showing most relevant terms (30) present in topic number 2 of depression dataset.

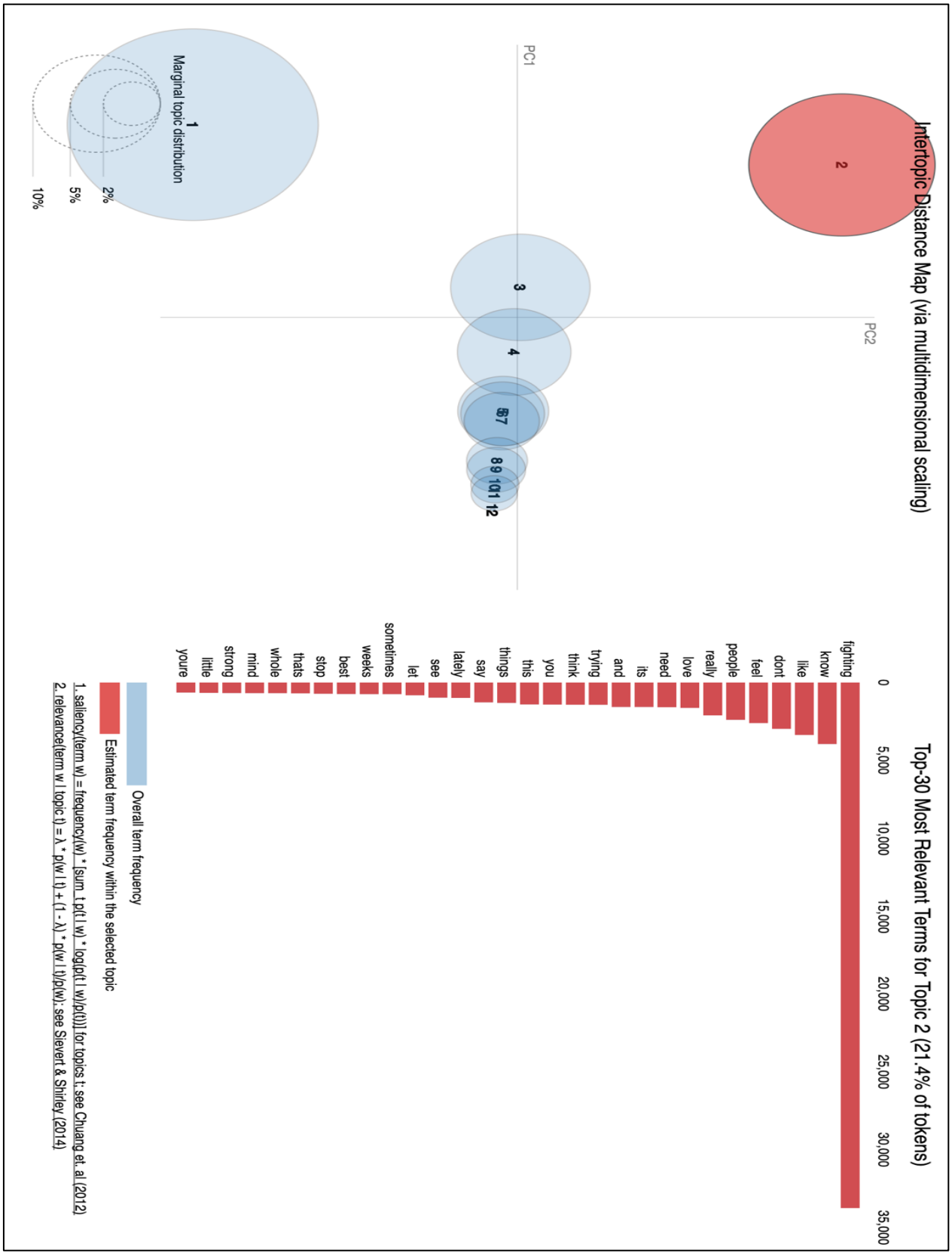
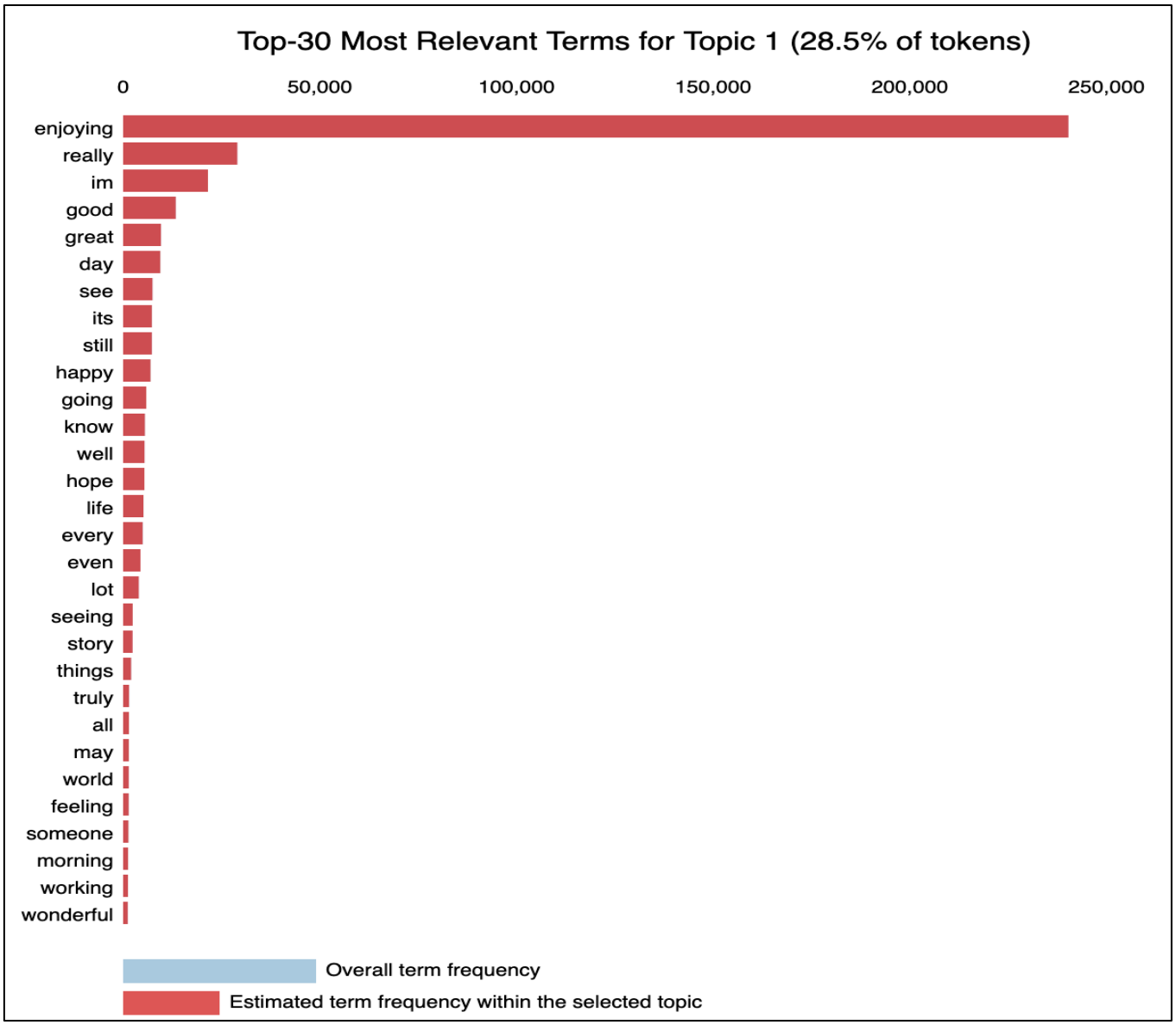
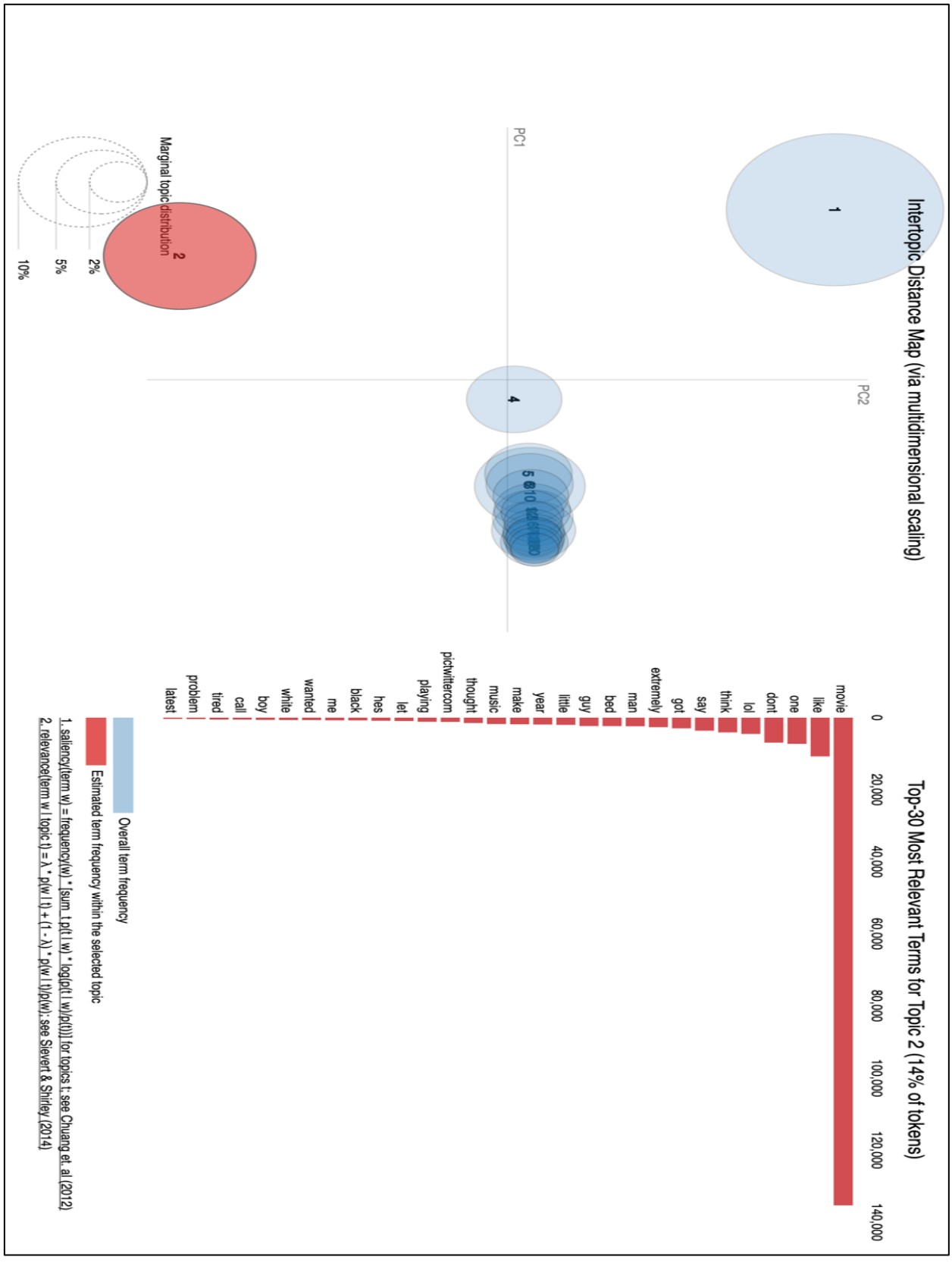


Illustration 6 Showing most relevant terms(30) present in topic number 1 of Non-depression dataset



The **illustration 6** depicts the dominance of words in topic one, when applied topic modeling to the non-depressed class. The results indicate that the top words in the topic one is enjoying, really, happy, great, good that indicate to the positive polarity. Similarly, the **Illustration 7** , portrays the dominant terms in topic two of non-depressed class. The frequent terms are movie, like, extremely, etc.

Illustration 7 Showing most relevant terms(30) in topic number 2 of Non-depression dataset



5.3.5 Information Retrieval (Conversion of Words to Vectors)

This section will discuss about the conversion of tweets into vectors by using two formulations, the first is by combination of TF-IDF and GloVe, and the second is by combination of Word2Vec and TF-IDF.

To initialize the processing of natural language in machine learning algorithms, the raw sentences needs to be converted into numeric format to serve as input to the algorithm. Each word in English language can be represented uniquely in numerical format. Each word is assumed to be distributed in n-space dimensions or vector space and each value of dimension has its own distinct value that adds weight to the meaning of the word in context of machine learning processing. One-hot encoding is one of the approaches to convert sentences to vector form. But this approach is just limited to assigning value '1' if word is present in the document otherwise '0'. This approach does not take other factors into consideration like the frequency of the word in the whole dataset, pertinence of specific word, semantics of the word and similarity of the word to the other words in the corpus.

Bag of Words: A method used to recapture information as well as for natural language processing techniques (NLP) under which a sentence is broken down into words with the removal of stop words, punctuation and grammar loading in a to z sequence. BoW helps to estimate the total number of words and their frequencies in a dataset [62]. Further, BoW helps to perform the topic dependent functions as well as using the set of most important words converted to vectors for input to Neural network learning techniques to perform better training for classification of data [62].

TF-IDF: TF-IDF stands for Term Frequency-Inverse Document Frequency. As the name indicates its relation to count frequency also its most vital purpose is to score the pertinency

of words in a document [62]. It is a statistic used to define the relevancy of the word within the document. It is a product of two terms, term frequency denoting the number of times the word appears in the sentence or the document and inverse document frequency, denoting the number of times the word is present in all the documents [69]. For instance, if the word is present in all documents then its TF-IDF score will be zero, which means, that word can be ignored and does not play an important part in defining distinct document. The TF-IDF is calculated by TF * IDF, where

Term frequency of a word = frequency of word in the document. [62][69]

Inverse document frequency of a word = $\log \frac{\text{number of documents}}{\text{number of times word appears in all documents}}$

GloVe Embeddings: For Natural Language Processing embeddings on words which are pre-trained are considered as most vital concepts under Deep Learning Techniques [70]. Further, the mindset of data scientist researchers considering Word2Vec embeddings as a predominant aspect under deep learning has been changed because of the increasing popularity of a new technique known as GloVe, due to its conscientious and more equitable procedure for word embeddings. The term GloVe stands for Global Vectors for word representation [70] which holds set of word vectors that can undergo an acquisition of transforming the connotation of words into a multi-dimensional trajectory. Moreover, GloVe implements on a universal layout of the documents to carry out each word count to retrieve as much as possible statistics on a comprehensive scale rather than investing time on local learning [70]. Besides, the parallel accompanying matrix helps the GloVe to acquire information and train on the words defined by vectors in such a formation that the co-occurrence ratios can be predicted by their deductions [70].

1. To develop a matrix P representing a structure where “a” is the framework of a word into vector space dimension of integrating the total frequency prevalence of its occurrence in compare to word “b”. However, the loss function shows the counterbalance of the words in ratio to their placement by using equation [74]:

$$Loss=1/balance$$

2. The calculation of impediments for respective word couple is defined as [74]:

$$M_a^t\, M_b + \, n_a + \, n_b = Log\, (P_{ab})$$

Where M_a Shows the preeminent word whereas the M_b shows the word used to convey the background situation. The values n_a and n_b are operated to prevent a biased outcome for scaling the values of the central as well as the words used for frame of context reference [74].

3. Operating Cost Optimization Function [74]:

$$\mathrm{C} \, = \sum_{a=1}^{x} \sum_{b=1}^{x} F(P_{ab})\, (M_a^t\, M_b + \, N_a + \, N_b - Log\, P_{ab})^2$$

The F is a balancing operator which behaves to control the model by avoiding from getting trained by using only intensely prevailing word couples.

4. Finally, a unified algorithm was developed as Global Vectors for Word Representation (GloVe) using the equation [74]:

$$F(P_{ab}) = \begin{cases} \left(\dfrac{P_{ab}}{p_{max}}\right)^{y} & \text{if } P_{ab} \; < \; PMAX \\ 1 & otherwise \end{cases}$$

Furthermore, for the application of semantically rich GloVe embeddings, we have considered pre-trained embeddings, of 50 dimensions, which means each word is presented in 50-dimensional space.

Steps for the combination of GloVe and TF-IDF:

1. Prepare the vocabulary of the dataset (i.e. splitting the whole pile of sentences into its constituent words)
2. Apply the TF-IDF vectorizer from the sklearn. The dimensions of the resultant matrix will be (the number of documents or sentences * length of the vocabulary), to exemplify, in our study, the representation for a matrix (A) taken as (119239 x 133547).
3. Load pre-trained GloVe embeddings
4. Compare each word in the prepared vocabulary with each word in the pre-trained GloVe embeddings file. If a word is present in the GloVe file, its n-dimension (50-dimension in our scenario) vector is extracted, otherwise, it is replaced with the vector of zeroes. This yields a matrix (B) represented as (133547 x 50) dimensions.
5. The next step is to project each document or sentence in a 50-dimensional vector space which is obtained by the dot product of matrices A and B. This will yield the output matrix of dimensions (119239 x 50), which will be fed into several statistical classifiers.

Word2Vec Embeddings: Word2Vec captures the input data by working under a cascading flow to memorize the pattern of words and train the embeddings of words in order to revamp the decay function by using gradient decent [70]. However, Word2Vec estimates the loss function by averaging the capability of a specific word to be able to anticipate its encircling words[70]. To exemplify, for a given phrase “The map and cap are in the car”.

For this phrase if the window of the background context is taken as 3 dimensions and the target term is set as “cap” then the contiguous words to cover the situation will be considered as “The” ,“map” ,“and” , “are” , “in” , and “the”. Further, these neighboring

words can be used to portray two type of approaches from Word2vec which include the continuous bag of words that targets to the forecast the word at center of attraction from the adjacent words in context as well as the skip gram technique that works as an unsupervised learning model to predict the words in context utilizing the word at point of convergence [70].

Furthermore, the Word2Vec embeddings used in this research are trained using the Gensim library. Gensim facilitates the feature to train embeddings based on customized datasets rather than already built embeddings. For the training of Word2vec embeddings, the dataset was passed in with the parameters that can be defined as [102]:

(i) size: it defines the number or the length of the dimensions in which each word [102] will be represented. It has been set to 50 in this research.

(ii) window: it defines the occurrences of the words that can be present between a specific word and other words around it. Has been set to 10 for this research.

(iii) min_count: it defines the number for which the word [102] has to be present in order to consider for training its embeddings. Has been set to 1 in this research.

(iv) workers: the number of threads dealing with the procedure. Has been set to 10 in the context of this research.

After training of Word2Vec embeddings, these are combined with TF-IDF by following the procedure similar to the GloVe embeddings:

Steps for the combination of Word2Vec and TF-IDF:

1. Apply the TF-IDF vectorizer from the sklearn. The resultant matrix will be of dimensions {number of documents or sentences * length of the vocabulary}, to exemplify, for a matrix (X) as (119239 x 133547) used in this study.

2. Load Word2Vec embeddings
3. Compare each word in the prepared vocabulary with each word in the word2Vec embeddings file. If a word is present, its n-dimension (50 in our scenario) vector is extracted, otherwise, it is replaced with the vector of zeroes. This yields another matrix (Y) as (133547 x 50) dimensions.
4. The next step is to project each document or sentence in a 50-dimensional vector space which is obtained by calculating the dot product of matrices (X) and (Y). This will yield the output matrix of dimensions 119239 x 50, which will be fed into several statistical classifiers.

Gensim: A method that is advertised as a bundle of functions performing autonomous topic modeling. However, in a pragmatic application its efficacy is much more than just performing topic modeling due to its cutting-edge technology and contemporary design that helps to produce text analysis by blending with all type of models used for vectorization of words such as GloVe, FastText, Neural Word and others to portray topic configuration pattern [84]. Into the bargain, gensim is the library facilitated by Python basically designed for text processing and supports the operations such as computing similarities, indexing of documents, and topic modeling [87]. The several characteristics of gensim includes that it considers all the attributes as independent attributes, such as not dependent on memory, and can support parallel computation for fast retrieval of results [87]. Nevertheless, it can also be easily applied to specific corpus and is flexible to enhance the dimensions of the training corpus. The word2vec embeddings are trained using gensim which organizes the textual content in each tweet into a 50-dimensional matrix. It helps in producing 50 possible related outcomes for each input trained by word2vec model using

gensim. For instance, if we input a word “depression” under the word2vec model, trained using gensim on the tweet level dataset, the output of possible related words under 50-dimensional matrix will be as shown in Illustration 8.

```
word_model_trained['depression']

/opt/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:1: Depre

Call to deprecated `__getitem__` (Method will be removed in 4.0.0, use se


array([ 3.2275705 , -3.9951835 , -7.910628  ,  0.79990286,  2.086765  ,
       -1.0862452 , -2.0619757 , -2.7626326 , -3.583358  , -1.8783809 ,
       -0.16882016,  2.7674413 ,  1.5225071 , -2.246026  , -2.6035514 ,
       -1.5751965 , -0.47910076,  7.161495  ,  3.7295492 , -1.9927654 ,
        1.1964264 ,  0.35512146, -1.469808  ,  1.763402  ,  0.84585416,
        4.7209296 ,  3.6430652 , -1.3263917 ,  3.320187  , -2.419976  ,
       -0.930503  , -0.5786733 , -0.44583938, -0.72394043, -0.24357943,
        5.241494  ,  0.16649947, -1.4612037 , -0.9576503 ,  3.8551853 ,
        0.7103843 , -2.9522245 ,  2.0365741 , -2.893442  , -0.26993838,
       -2.677091  , -2.4280322 , -2.3115888 ,  1.3575921 ,  1.5714914 ],
      dtype=float32)
```

Illustration 8 showing matrix for trained Word2Vec model

Furthermore, let’s take an example of a tweet to represent the background process of a matrix formed under combination of GloVe + TF-IDF as well as Word2Vec + TF-IDF. The output matrix from each of these two combinations will be the input for the classification models. The transformation of a tweet under each case is described as following:

A. Combination of Glove + TF-IDF:

Stage 1. Matrix for TF-IDF: After the data cleaning process, the TF-IDF weight of each term in a Tweet is calculated and stored under a sparse matrix. For this example, if we take a tweet with 21 terms, then the resulting TF-IDF weight of each term in the Tweet will be as shown under Illustration 9

```
S = csr_matrix(tweets_matrix[4])
print("TF-IDF Sparse matrix: \n",S)
```

```
TF-IDF Sparse matrix:
   (0, 84)       0.19482918688568843
  (0, 83)        0.18653892178973389
  (0, 82)        0.23791154328965167
  (0, 81)        0.18006553980318563
  (0, 80)        0.24809239520162366
  (0, 79)        0.21095900517292746
  (0, 78)        0.2223945316167524
  (0, 77)        0.2274897023789461
  (0, 76)        0.3899375571172546
  (0, 75)        0.25439073382428956
  (0, 74)        0.22088080456908007
  (0, 73)        0.2511648123774251
  (0, 72)        0.26121314710552956
  (0, 71)        0.16123239985840565
  (0, 70)        0.1886690235148852
  (0, 69)        0.21148435108764985
  (0, 47)        0.10710823439777696
  (0, 16)        0.1463192528077864
  (0, 4)         0.19172200607868473
  (0, 3)         0.15774018123631892
  (0, 1)         0.18233549803819898
```

Illustration 9 showing sparse matrix for TF-IDF

Stage 2. Matrix for GloVe Embeddings: The same Tweet under GloVe embeddings is treated as a single bucket with 21 terms. Further, the weight of the whole bucket is calculated by comparing each term in the tweet with the pre-trained Glove Embeddings. If the term matches with the pre-trained GloVe embeddings, it's 50-dimension vector gets extracted. The output Matrix for the Tweet into 50-Dimensional vector space is as shown in illustration 10

```
S = csr_matrix(word_matrix_embeddings[4])
B = S.todense()
print("glove embeddings: \n", B)
```

```
glove embeddings:
 [[ 0.39913  -0.81229   0.32291  -0.28448  -0.11158   0.89293  -0.21247
   0.81566  -0.29148   0.22767  -0.11988   0.26219  -0.8213   -0.12922
   0.061769 -0.51359   0.052784 -0.61688  -0.8976    0.047951  0.87656
   0.15529  -0.2125   -0.16023  -0.17968  -1.9219    0.3067   -1.3344
   0.69431  -0.36132   1.6507   -0.10313  -1.199    -0.85255   0.40578
   0.35499  -0.028773 -0.46963  -0.38668   0.22262  -0.20836   0.08044
   0.21401  -0.14683   0.23843  -0.69019   0.11019  -0.46148   0.5094
  -0.61885 ]]
```

Illustration 10 showing matrix for GloVe embeddings

Stage 3. Matrix for GloVe and TF-IDF: The final output for the same Tweet is then processed by taking the dot product of both the metrices from stage 1 and stage 2 under

case A. Ultimately, the 50-dimension matrix obtained is the result of the Tweet using combination of GloVe and TF-ID as shown in illustration 11.

```
S = csr_matrix(final_tweets_matrix[4])
B = S.todense()
print("Glove + TF-IDF: \n", B)
```

```
Glove + TF-IDF:
 [[ 1.25929629 -0.52772086  0.26553106 -0.88924132  1.48117615  1.58333025
   0.10266026  0.61995461 -0.74398302  0.81924042  0.27568542  0.19482446
  -1.24071375 -0.12049306  2.620444    0.98723346 -0.58410238  0.12755896
  -0.66485377 -0.92198211 -0.59994146  1.42529565  1.52443677  0.96042091
   1.88171137 -5.46020142 -1.95136148 -0.31837014  2.82439774 -0.17958054
   9.8149568   0.84231878  0.02621819 -2.27593327 -0.42911314  0.52908213
   0.21392679  0.33798043  0.42090031 -0.79224577 -1.03808571 -0.38397487
   0.31529404  2.53482498  1.44734205 -0.18928294  0.06641524 -0.45810336
   0.58392409  0.69100639]]
```

Illustration 11 showing matrix for GloVe and TF-IDF

B. Combination of Word2Vec + TF-IDF:

Stage 1. Matrix for TF-IDF: Considering the same example for a Tweet with 21 terms, the output of stage 1 under case B is same as the output of stage 1 under case A.

Stage 2. Matrix for Word2Vec Embeddings: Unlike the pre-trained GloVe embeddings, the Word2Vec embeddings are trained on the Tweet level dataset using gensim. Further, using the same Tweet with 21 terms the word2vec embeddings are also calculated. The whole Tweet is treated as a single bucket. Further, the weight of the bucket is calculated by extracting word2vec embeddings in 50-dimension for each term in the tweet as shown in Illustration 12.

```
S = csr_matrix(word_vector_embeddings[4])
B = S.todense()
print("word2vec embeddings: \n", B)
```

```
word2vec embeddings:
 [[ 5.9734664   0.6310016  -4.588813    1.651538    1.9868397  -1.7386596
  -4.120953    3.402071    0.5155075  -3.7933753  -6.065419    5.028276
  -2.981268   -4.295137   -1.0704504  -0.8749166  -3.256247    1.9120423
   0.7522198   3.6496274   0.6942794  -1.2072089  -3.4863591   0.04962726
   1.0564413   4.179343   -3.659727   -2.9100568  -1.8187703   0.97587985
  -0.01675744 -1.1130638  -0.09645098 -0.940086    1.9391406   4.739372
  -6.8861837   2.2638113  -1.9229968  -0.08238193 -2.855467   -2.555551
   0.25767124 -0.48622692 -2.2049894  -1.6723586  -1.7160431  -2.2270513
  -0.22170933  2.6303535 ]]
```

Illustration 12 showing matrix for Word2Vec embeddings

Stage 3. Matrix for Word2Vec and TF-IDF: The Final output for the same Tweet with 21 terms is then processed by taking the dot product of both the metrices from stage 1 and stage 2 under case B. Ultimately, the 50-dimension matrix obtained is the result for a Tweet using combination of Word2Vec and TF-ID as shown in illustration 13

```
S = csr_matrix(wordvec_tweets_matrix[4])
B = S.todense()
print("Word2Vec + TF-IDF: \n", B)
```

```
Word2Vec + TF-IDF:
 [[ 7.87469511e+00 -1.04000241e-02 -7.38251764e+00  4.77479123e+00
  -2.60233241e+00 -7.59221257e+00 -4.46633861e+00 -3.44057748e+00
  -3.83426870e+00 -7.89076490e+00 -4.69695249e+00  4.66800610e+00
  -1.42984823e-01 -5.54260345e+00 -3.37752421e+00  2.76564428e-01
  -2.71503282e+00  8.81527308e+00  3.29449745e+00 -2.21849759e+00
   6.67032180e+00  1.61444304e+00 -3.63329160e+00 -1.11230333e-01
   4.92366462e+00  1.10869119e+01  1.40234342e-01  1.18400165e+00
  -5.96162088e+00 -6.55846704e-01 -2.49411839e-01  3.33001205e-01
   1.20751472e+00 -7.53663000e-01  3.75606640e+00  3.53656636e+00
  -2.30646660e+00 -3.45631277e-01 -2.62015168e+00  7.36177286e-01
   1.78316478e-01  2.05779078e+00  2.86240539e+00 -1.78752149e+00
  -2.61867751e+00 -3.21516585e+00 -2.22346394e+00 -2.62111850e+00
  -6.05090551e+00  3.35431103e+00]]
```

Illustration 13 showing matrix for Word2Vec and TF-IDF

Nevertheless, the final outputs in stage 3 under both the cases A and B are then used as separate inputs for each of the classification models. Similarly, the output matrix for each tweet under Tweet Level dataset is calculated using both the combinations of GloVe and TF-IDF as well as Word2Vec and TF-IDF. Finally, these output matrices from each combination are separately operated into all machine learning models discussed under section 5.4 using a training and testing ratio of 70% and 30% respectively.

5.4 Advanced Machine and Deep Learning Algorithms by Using Ensemble Learning Methods

To begin with, sometimes the decision-making process gets so byzantine that we need our trusted ones to show a point of view from their own speculation in order to ward off any tendentious outcome [29]. Similarly, in the case of machine learning when a model performs a task individually there exists odds where a lone model gets discrepancy or fluctuation with a partial outcome [29]. Hence, necessitating the handling of ensemble learning.

Moving further, Ensemble Methods are delineated as a machine learning technique that gravitates to combine a number of peculiar models using the identical algorithm learning with the purpose of bringing forth a sole gilt-edge prognosticative configuration that produces an output model which is nonpartisan and shows less deviation towards data [29]. However, this architecture further comprises three meta-algorithms which are broken down into bagging, boosting as well as stacking.

Stacking: A consolidation of multifarious weak learning models learn in parallel and then a meta-model is trained to combine them for producing a better prediction output in reliance on the results obtained from weak predictions by those divergent weak learning models [29].

Bagging: Instead of treating the dataset as a whole, it is breakdown into subsets that are trained by different models individually in a coextending manner and integrate to Produce output with less discrepancy. [29]. Moreover, bagging is involved in the random forest working as an ensemble technique model whereas it works as an individual model in case of the decision tree [29].

Decision tree: A tree-like structure under which the decision-making process is rendered in the branches and a characteristic aspect is portrayed by each node whereas the aftermath value is revealed as the leaf. [33]

Gini Index, Information gain as well as the theory of entropy are utilized as preeminent measures to select an attribute provided to the root node [50][33].

Random Forest Classifier: A structure of multiple decision trees gets developed after training. Further, the combination of groups of these trees is referred to as a forest. Identical to the procedure of decision trees these groups of trees are spawn using a rule-based routine. The dataset gets its target value predicted by apportioning into smaller values until the resulting nodes are achieved also called the best-split method [31][32]. The testing features are handled to generate a predicted outcome that is stored and further calculates the rank of each predicted outcome. The model considers the prediction which has the highest rank. The significance of each feature is dependent on the normalized ratio calculated by the division of the sum of splits of a specific node to the sum of the total number of nodes [31][32]. Library scikit-learn determines the importance of nodes via Gini Index, making an assumption for the existence of binary tree

$$Bi_n = wg_n * D_n - wg_{left(n)} * D_{left(n)} - wg_{right(n)} * D_{right(n)} \quad [31][32]$$

Where

Bi_n = the importance of node n

Wg_n = number of weighted samples arriving node n

D_n = numeric value indicating the purity of node n

Left(n) = left child node obtained from left split on node n

Right(n) = right child node obtained from right split on node n

After computation, the importance of each feature is calculated and normalized to retain their values between magnitude of 0 and 1.

The feature importance in Random Forest at last stage is calculated by:

$$F(i) = \frac{\text{(sum of feature importance value on every tree)} * \text{(normalized features)}}{\text{total number of trees}}$$ [31][32]

Where F is the feature and (i) represents the importance.

Table 16 Showing Pseudo Code for Random Forest Algorithm

```
1. #Defining the function
   r_f=RandomForestClassifier ()
2. #Runtime complexity
   start_time=time. time ()
3. #Fitting the model for training
   r_f.fit (X_train, Y_train)
4. #Predicting the outcome
   value_pred_r_f = r_f. predict (X_test)
5. #Applying K-fold cross validation
   scores_r_f=cross_val_score (r_f, X_test, Y_test, cv=10)
6. #Scaling time complexity
   print ("Finished in ... “, time. time()-start_time)
7. #Printing confusion matrix
   print ("Confusion Matrix obtained for random forest is \n”,
   confusion_matrix (Y_test, value_pred_r_f))
8.  #Printing results
   print ("Accuracy after applying random forest”, (accuracy_score (Y_test,
   value_pred_r_f)))
   print ("Scores after applying 10 CV on random forest”, np.
   average(scores_r_f))
   print ("Precision for random forest “,
   precision_score(Y_test,value_pred_r_f))
   print("Recall for random forest ",recall_score(Y_test,value_pred_r_f))
```

Logistic regression: A classification technique designed to select distinct classes in a set and allocating observations to each selected set making it distinguishable to linear regression where persistent numerical values are released as output [60]. What is more, a sigmoid function is operated under Logistic regression which helps to generate a

probability estimate that works within the calibration of two or more divergent sets of classes [60]. The algorithm of logistic regression outputs only two values (dichotomy), depending on the usage of one or more predicting factors [77]. It can be considered as the special scenario of linear regression, where the output belongs one category, but obtained by applying logarithmic function on the predicted probability. The prediction equation from [78] for the logistic regression for two variables can be defined as:

$$\text{Output} = \exp(a0 + a1 * x) \;/\; (1 + \exp(a0 + a1 * x)) \qquad [78]$$

Table 17 Showing Pseudo Code for Logistic Regression Algorithm

```
1. #Defining the function
   load_log_reg=LogisticRegression(penalty='l2',random_state=None)
2. #Runtime complexity
   start_time=time.time()
3. #Fitting the model for training
   load_log_reg.fit(X_train,Y_train)
4. #Predicting the outcome
   value_pred_log = load_log_reg.predict(X_test)
5. #Applying K-fold cross validation
   scores_log_reg=cross_val_score(load_log_reg, X_test,Y_test, cv=10)
6. #Scaling time complexity
   print("Finished in ... ",time.time()-start_time)
7. #Printing confusion matrix
   print("Confusion Matrix obtained for Logistic Regression is
   \n",confusion_matrix(Y_test,value_pred_log))
8. #Printing results
   print("Accuracy after applying Logistic Regression is",
   (accuracy_score(Y_test,value_pred_log)))
   print("Scores after applying 10 CV on Logistic
   Regression",np.average(scores_log_reg))
   print("Precision for Logistic Regression ",precision_score(Y_test,value_pred_log))
   print("Recall for Logistic Regression ",recall_score(Y_test,value_pred_log))
```

SVM: A support vector machine is defined as a classifier with discriminative nature helping to justify outputs separated by hyperplane according to the training of a predefined data. SVM is used to solve regression as well as classification problems where basically

the target is to form a split section which helps to divide the labeled vectors by an opinion boundary also called a decision line. [61]. In classification algorithm of SVM, a linearly defined hyper-plane can be obtained between two distinguished classes. SVM algorithm utilizes the SVM kernel function, called the kernel trick [79]. The SVM kernel takes input in lower dimensions and projects into high dimensional space. This type of approach is convenient for non-linear separable problem. The loss function computed in SVM kernel is hinge loss function, and the loss function for SVM can be calculated by equation adapted from [79] shown as:

$$\text{Loss Function for SVM} = \quad \text{Min}_{wg}\ ß\ ||\ wg\ ||^2 + \sum_{i=1}^{d} (\ 1 - \ o_i(i_i\ ,\ \ wg)$$

Table 18 Showing Pseudo Code for SVM

```
1. #Defining the function
    load_svm=svm.SVC(C=1.0, kernel='rbf', degree=3)
2. #Runtime complexity
    start_time=time.time()
3. #Fitting the model for training
    load_svm.fit(X_train,Y_train)
4. #Predicting the outcome
    value_pred_svm = load_svm.predict(X_test)
5. #Applying K-fold cross validation
    scores_svm=cross_val_score(load_svm, X_test,Y_test, cv=10)
6. #Scaling time complexity
    print("Finished in ... ",time.time()-start_time)
7. #Printing confusion matrix
    print("Confusion Matrix obtained for Support Vector Machine Classifier is
    \n",confusion_matrix(Y_test,value_pred_svm))
8. #Printing results
    print("Accuracy after applying Support Vector Machine
    is",(accuracy_score(Y_test,value_pred_svm)))
    print("Scores after applying 10 CV on Support Vector Machine",np.average(scores_svm))
    print("Precision for Support Vector Machine" ,precision_score(Y_test,value_pred_svm))
    print("Recall for Support Vector Machine ",recall_score(Y_test,value_pred_svm))
```

Boosting: It helps to frame a strong predictive model working in a sequential iterative fashion by learning the incorrect classified observations from the previous model and fine-tune its weight for the input to the next model in such a way to yield less biased results [29].

Gradient boosting: A methodology under which the gradient descent algorithm is operated under models that are working in a follow-up subsequent manner by curtailing errors in each model at the same time [51]. Moreover, it can be defined as a machine learning boosting technique that helps to reduce the error in the chiefly predicted global output by integrating the previous models with the subsequently most outshine feasible model [52]. However, to diminish the error from the upcoming model a principal target is stipulated justified by error's gradient in the direction of prediction [52]. The main benefits for using XGBoost as a classifier are the regularization parameter and dealing with sparse matrices. Implementation of gradient Boosting involves the following steps:

$$\mathrm{Fun(v)} = argmin_{\beta} \sum_{k=1}^{d} \mathrm{Loss}(w_i, \beta) \quad [71]$$

where the loss of the function is calculated in iterative manner

The boosting model can be defined for each end node by using:

$$\mathrm{Fun_j(v)} = \mathrm{Fun_{j-1}}\ (v) + ß_j h_j(x) \quad [71]$$

Table 19 Showing Pseudo Code for Gradient Boosting Algorithm

```
1. #Defining the function with decision tree as default weak learner
   load_gdb = GradientBoostingClassifier(n_estimators=100)
2. #Runtime complexity
   start_time=time.time()
3. #Fitting the model for training normalized features
   load_gdb.fit(X_train,Y_train)
4. #Predicting the outcome
   value_pred_gdb = load_gdb.predict(X_test)
5. #Applying K-fold cross validation
   scores_gdb_=cross_val_score(load_gdb, X_test,Y_test, cv=10)
6. #Scaling Round off value
   gdb_predictions = [round(value) for value in value_pred_gdb]
7. #Compute time complexity
   print("Finished in ... ",time.time()-start_time)
8. #Printing confusion matrix
   print("Confusion Matrix obtained for Gradient Boosting is
   \n",confusion_matrix(Y_test,gdb_predictions))
9. #Printing results
   print("Accuracy after applying Gradient
   Boosting",(accuracy_score(Y_test,gdb_predictions)))
   print("Scores after applying 10 CV on Gradient Boosting",np.average(scores_gdb_))
   print("Precision for Gradient Boosting ",precision_score(Y_test,adb_predictions))
   print("Recall for Gradient Boosting ",recall_score(Y_test,adb_predictions))
```

ADABOOST: It is one of the supervised machine learning algorithms that operate on the sequential aggregation of the feeble classifiers, by learning from the weighted training data [54]. The weak learners assemble as a bunch and train to output a robust decision. The model forecasts the values by computing the weighted average of all the deficient classifiers. It works by learning the significant and dominant data samples more prominently in an iterative manner than the samples that do not require much overhead. The procedure of estimation continues relentlessly until the error rate of estimation drops to the maximum extent or till the training data yields minimum classification error [55]. Working of AdaBoost requires assignment of weight to each training data item. The item

which has been wrongly predicted will be assigned larger weight value so that after next iteration, the probability of predicting those wrong values increases. The AdaBoost model workings is based on:

$$\text{Out(s)} = \text{sign}(\sum_{k=1}^{K} \emptyset_k o_k(s)) \qquad [73]$$

Where, o(s) = the output of a weak classifier k for data item input s

Theta_k = numerical value indicating weight assigned to the classifier

Table 20 Showing Pseudo Code for AdaBoost Algorithm

```
1. #Defining the function with Random Forest as weak learner
   load_adb = AdaBoostClassifier(base_estimator= r_f, learning_rate = 0.01)
2. #Runtime complexity
   start_time=time.time()
3. #Fitting the model for training
   load_adb.fit(X_train,Y_train)
4. #Predicting the outcome
   val_pred_adb = load_adb.predict(X_test)
5. #Applying K-fold cross validation
   scores_adb_=cross_val_score(load_adb, X_test,Y_test, cv=10)
6. #Round off value
   adb_predictions = [round(value) for value in val_pred_adb]
7. #Scaling time complexity
   print("Finished in ... ",time.time()-start_time)
8. #Printing confusion matrix
   print("Confusion Matrix obtained for AdaBoost is
   \n",confusion_matrix(Y_test,adb_predictions))
9. #Printing results
   print("Accuracy after applying
   AdaBoost",(accuracy_score(Y_test,adb_predictions)))
   print("Scores after applying 10 CV on
   AdaBoost",np.average(scores_adb_))
   print("Precision for AdaBoost ",precision_score(Y_test,adb_predictions))
   print("Recall for AdaBoost ",recall_score(Y_test,adb_predictions))
```

XG Boost: An algorithm extensively designed to extend the limitations of the computational power of tree algorithms [53] and possesses the framework of gradient

boosting. It can disentangle diverse problems ranging from classification, regression, ranking and customized prediction problems [52]. It constructs the trees by adopting parallelized pursuits and prunes the trees using a depth-first approach [52]. It optimally utilizes the hardware assets to deal with huge data frames and has inbuilt methods of cross-validation that specifies the number of iterations required by the boosting algorithm [52]. The loss function along with regularization (objective function) at each iteration (i), that is required to be reduced by using equation adapted from [72]:

$$\text{Loss\_reg}^{(i)} = \sum_{k=1}^{s} loss(a_k , y_k^{(i-1)} + \text{fun}_i (j_k)) + Ø (\text{fun}_i)$$

Where, a_k = actual label of the value, y_k = predicted label

fun (j) = linear approximation of a function around point j

Ø(fun) = second-order Taylor approximation of a function

Table 21 Showing Pseudo Code for XGBoost Algorithm

```
1. #Defining the function with random forest as weak learner
   load_xg_boost = xgb.XGBRegressor(gamma=0.1,base_estimator= r_f,
   objective='binary:logistic', learning_rate=0.01, max_depth=10, n_estimators=200,
   random_state=1)
2. #Fitting the model for training
   load_xg_boost.fit(X_train, Y_train)
3. #Predicting the outcome
   val_pred_xg_boost = load_xg_boost.predict(X_test)
4. #Applying K-fold cross validation
   scores_xg_boost=cross_val_score(load_xg_boost, X_test,Y_test, cv=10)
5. #Scaling Round off value
   xg_boost_predictions = [round(value) for value in val_pred_xg_boost]
6. #Printing confusion matrix
   print("Confusion Matrix obtained for XG Boost is
   \n",confusion_matrix(Y_test,xg_boost_predictions))
7. #Printing results
   print("Accuracy after applying XG
   Boost",(accuracy_score(Y_test,xg_boost_predictions)))
   print("Scores after applying 10 CV on XG Boost",np.average(scores_xg_boost))
   print("Precision for XG Boost ",precision_score(Y_test,xg_boost_predictions))
   print("Recall for XG Boost ",recall_score(Y_test,xg_boost_predictions))
```

Artificial Neural Networks (ANN): A machine-based neural network synthesized to solve prediction analysis and categorical classification where the brain like stimulations are being framed to create an artificial network of neurons working with deep learning concepts, identical to the biological reticulum and chemical reactions transpiring in a human brain.

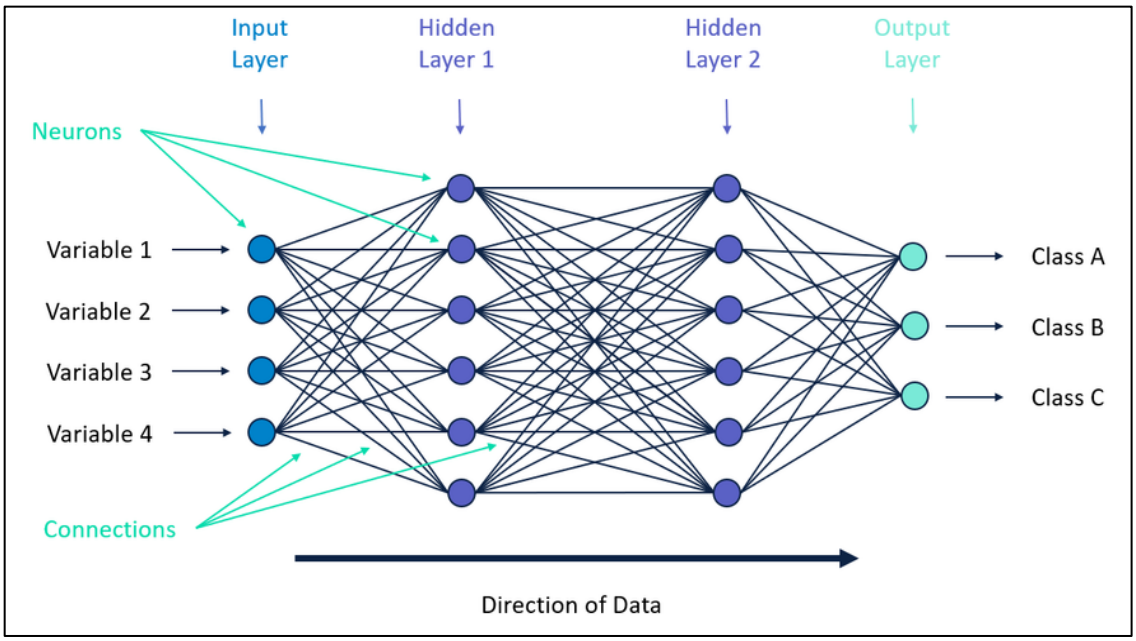


Illustration 14 Showing Neural Network (Credits [76])

Quadratic Cost Function can be calculated for Neural networks by using equation adapted from [75] can be represented as :

$$CF = 1/2 \left(\sum_k (P_k - A_k^N)^2 \right)$$

Where,

P_k = Predicted value

A_k = Activation Function on the input

CF = Cost function for output activations

Recurrent Neural Networks: RNN works on the substratum of perceptron's unsettling in nature. Moreover, they work together within a model by a combination of timely connections and connectivity between the passes [57]. They can store a much larger set of instructions gathered from the effectiveness of events bygone [57]. However, workflow with a non-linear gesture permits various convoluted measures in order to modify the hidden layer for updating [57]. Nevertheless, Recurrent neural networks can produce various sophisticated sequels using a bulk of small chunks storing information units passing into programs running alongside resulting unique new transformations.

Sequential Neural networks: A machine learning technique galvanized by the working concept of a human brain perception to acquire information from the empirical data [56]. To add to it, making a computer working into a human-like brain thinking program solver is a perplexing task but breaking down the concepts on which the nervous system gets in action to give a signal through neuron cells helps to develop a neural network algorithm that can be recognized by a computer. Further, the neural network works in three layers beginning with the raw information being passed within the input layer moving to the next layer to be analyzed called as the hidden layer where learning and processing of the data using labyrinthine computation calculations are done and lastly, carried to the output layer to reveal the investigated outcome[56].

LSTM: The full form of LSTM is long short-term memory [59]. As indicated through its name the function of LSTM is quite similar in relation to memory increment under the case of RNN where there is less space to hold the event information from the previous instance that will ultimately be used as an asset for improvisation in the learning process of the current neural network [59]. LSTM is utilized in various fields such as deviation of time

series, helps in sound engineering for music composition, recognition of voice commands as well as learning and identification of handwriting patterns [59].

Table 22 Showing Pseudo Code for LSTM

```
#loading lstm for sequential classification
    model_lstm_load = Sequential()
# Embedding_Dimensions
model_lstm_load.add(Embedding(len(word_matrix_embeddings),EMBEDDING_D
    IMENSIONS,weights=[word_matrix_embeddings],input_length=10,
    trainable=False))
#Setting other parameters
    model_lstm_load.add(Conv1D(kernel_size=3, filters=32,
    activation='relu',padding='same'))
#adding pool size
    model_lstm_load.add(MaxPooling1D(pool_size=2))
#selecting parameter for dropout
    model_lstm_load.add(Dropout(0.2))
#add layer
    model_lstm_load.add(LSTM(300))
#adding sigmoid parameter
    model_lstm_load.add(Dense(1, activation='sigmoid'))
#defining number of epochs
    EPOCHS=5
#defining early stop
    early_stop = EarlyStopping(monitor='val_loss', patience=3)
#Fitting and training the model
    model_seq_hist = model_lstm_load.fit(x_train, y_train,epochs=EPOCHS,
    batch_size=12, shuffle=True,callbacks=[early_stop])
```

BI-LSTM: The model of BI-LSTM or bidirectional long short-term memory is defined as an algorithm based on the framework of two separate Recurrent Neural Networks. It works on the concept of learning events in step by step layers of time series platform. It can also be trained to learn layers of events in any sequence from past to future as well as future to past, ultimately building a powerful learning network that boosts up the retaining performance for perpetuating information to be utilized anytime by blending events from twain layers of time framing whole of the backward and forward trials.

Table 23 Showing Pseudo Code for Bi-Lstm Algorithm

```
#creating check of maximum words
    MAX_NUM_WORDS_check = len(counts)
#tokenizing context
    tokenizer_check= Tokenizer(num_words=MAX_NUM_WORDS_check)
    tokenizer_check.fit_on_texts(cleaned_tweets)
    word_vector_obtained = tokenizer_check.texts_to_sequences(cleaned_tweets)
    word_index_obtained = tokenizer_check.word_index
#checking size of vocab
    vocab_size_check = len(word_index_obtained)
#printing vocab for display
    vocab_size_check
#defining sequence length
    MAX_SEQ_LENGTH_CHECK = 10
#creating input tensor
    input_tensor_obtained = pad_sequences(word_vector_obtained,
    maxlen=MAX_SEQ_LENGTH_CHECK)
    print(input_tensor_obtained.shape)
#defining shape of embeddings
    EMBEDDING_DIMENSIONS = 50
#creating embedding matrix
    embedding_matrix = np.zeros((MAX_NUM_WORDS_check,
    EMBEDDING_DIMENSIONS))
#initializing neural nets
    input_obtained = Input(shape=(MAX_SEQ_LENGTH_CHECK,))
#Embedding_Dimensions
    x_obtained = Embedding(MAX_NUM_WORDS_check,
    weights=[word_matrix_embeddings])(input_obtained)
    x_obtained = Bidirectional(LSTM(10 ,
    recurrent_dropout=0.01,return_sequences=True,dropout=0.20))(x_obtained)
#preparing inputs
    x_obtained = GlobalMaxPool1D()(x_obtained)
#applying relu effect
    x_obtained = Dense(10, activation="relu")(x_obtained)
    x_obtained = Dropout(0.20)(x_obtained)
#applying sigmoid reg
    x_obtained = Dense(1, activation="sigmoid")(x_obtained)
#fitting neural nets on data
    model_lstm.fit(x_train, y_train, batch_size=8, epochs=5)
```

5.5 Computation of algorithms using various metrics

Python library provides the feature of application of various classification algorithms whose performance and accuracy results can be tested by employing various metrics such as confusion matrix, accuracy, precision, recall(sensitivity), specificity. Confusion matrix yields the output in the context of four terms True Positive, True Negative, False Positive, False Negative [98]. These terms can be defined as following:

	Actual	Actual
Predicted	True Positive	False Positive
Predicted	False Negative	True Negative

Table 24 (a) Showing Confusion Matrix ([104][105])

True Positive (TP): The total number of correctly classified positive class labels (For example, the model predicted “yes”, and the actual value is “yes”)

True Negative (TN): The total number of correctly classified negative class labels (For example, the model predicted “No”, and the actual value is “No”)

False Positive (FP): The total number of class labels that are negative in actual but predicted positive (For example, the model predicted “Yes”, but the actual value is “No”)

False Negative (FN): The total number of class labels that are positive in actual but predicted negative (For example, the model predicted “No”, but the actual value is “Yes”)

Further, the other metrics such as accuracy, precision, recall are calculated based on True Positive, True Negative, False Positive, False Negative output fields, which are analyzed by using the following formulas.

Table 24 (b) Showing evaluation of Confusion Matrix using various metrics (Credits [104][105])

MEASURE	DERIVATION
SENSITIVITY(RECALL)	TPR = TP / (TP + FN)
SPECIFICITY	TNR = TN / (FP + TN)
PRECISION	PPV = TP / (TP + FP)
NEGATIVE PREDICTIVE VALUE	NPV = TN / (TN + FN)
FALSE POSITIVE RATE	FPR = FP / (FP + TN)
FALSE DISCOVERY RATE	FDR = FP / (FP + TP)
FALSE NEGATIVE RATE	FNR = FN / (FN + TP)
ACCURACY	ACC = (TP + TN) / (TP + TN + FP + FN)
F1 SCORE	F1 = 2TP / (2TP + FP + FN)
MATTHEWS CORRELATION COEFFICIENT	MCC = TP*TN - FP*FN / SQRT((TP+FP)*(TP+FN)*(TN+FP)*(TN+FN))

Sensitivity: It is also known as **Recall** value which shows the completeness capability of the model to analyze all possible compatible Instances as True Positive Rate [104][105].

Specificity: It is also known as the value for True Negative Rate which shows the ratio of real negatives that gets analyzed correctly [104][105].

Precision: It is also known as the Positive Predictive Value which shows the correctness capability of the model to produce only the possible significant occurrences [104][105].

Negative Predictive Value: It is the value that shows the correctness of probability towards the outcome of predicted negatives that are true negative values in real [104][105].

False Positive Rate: It is shown by the ratio of false positives to the total value of real negative instances (FP + TN) where false positives are those values which are negative in real but predicted outcome is positive [104][105].

False Negative Rate: It is shown by the ratio of false negatives to the total value of real positive instances (FN + TP) where false negatives are those values which are positive in real but predicted outcome is negative [104][105].

False Discovery Rate: It is shown by the ratio of false positives to the total value of all predicted positive outcomes (FP + TP) [104][105].

Accuracy: It is an important rating operated when both prediction classes are of same importance. It is the ratio of sum of predicted cases that are identified correctly (TP + TN) to the total number of predicted outcomes (TP + TN + FP + FN) [104][105].

F1- Score: It is represented by the harmonic average of Precision and Recall [104][105]. It is used to identify the offset within Recall and Precision value to counterbalance the equation using weights.

Matthews correlation coefficient: It helps to understand the relation between the real values and the predicted outcomes. It generates a large value only when both the prediction outcomes for negative occurrences and the positive occurrences are accurately classified with a high percentage. The higher the value of MCC, the more is the quality of predictive model.

Receiver Operating Characteristics curve (ROC): It is used to testify the performance of the classification algorithm by setting various parameters of thresholds [96]. Further,

ROC curve is a probabilistic curve, depicting the probabilities of correctly classified values and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model has performed on the problem of separation. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes. For instance, the value from 0.7 to 0.8 shows acceptable classifier[106], the values from 0.8 to 0.9 shows excellent classifier[106], the values above 0.9 shows outstanding classifier[106] and the value 1 represents a perfect classifier [106].

Normalized Confusion Matrix: A normalized confusion matrix is used to provide an easy way of understanding the outcomes in a decimal value. The term Normalized represents that each row has a counterbalanced sum of values as 1.00 which shows that sum of each row makes 100% outcome for the classification of instances in a class. The higher the value in the gradient diagonal, the higher is the degree of accurately predicted outcomes. To add to it, the off-diagonal values which are incorrectly predicted outcomes, leads to create the confusion in a matrix because they were mixed up by error with another class.

5.6 Results of All Advanced Machine and Deep Learning Algorithms for Tweet Level Architecture

For the following experimental results, classification models are applied on both type of word embeddings i.e. combination of TF-IDF and GloVe, as well as the combination of Word2Vec and TF-IDF. Below are the results of various machine learning algorithms implemented on both types of embeddings.

5.6.1 Results of Random Forest Using GloVe +TF-IDF for Tweet Level:

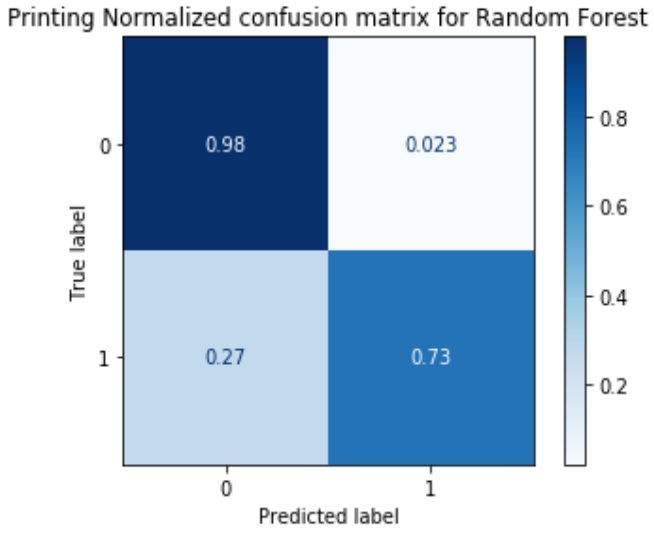


Illustration 15 Showing Normalized Confusion Matrix for Random forest

The Illustration 15 shows a normalized confusion matrix for Random Forest using GloVe +TF-IDF for Tweet Level. The decimal representation helps to understand the results by observation of the diagonal values showing the degree of accurately predicted occurrences, which are higher than the off diagonal values. Moreover, the sum of values of each row is equal to 1.00 that shows the 100% utility of possible outcomes for each class.

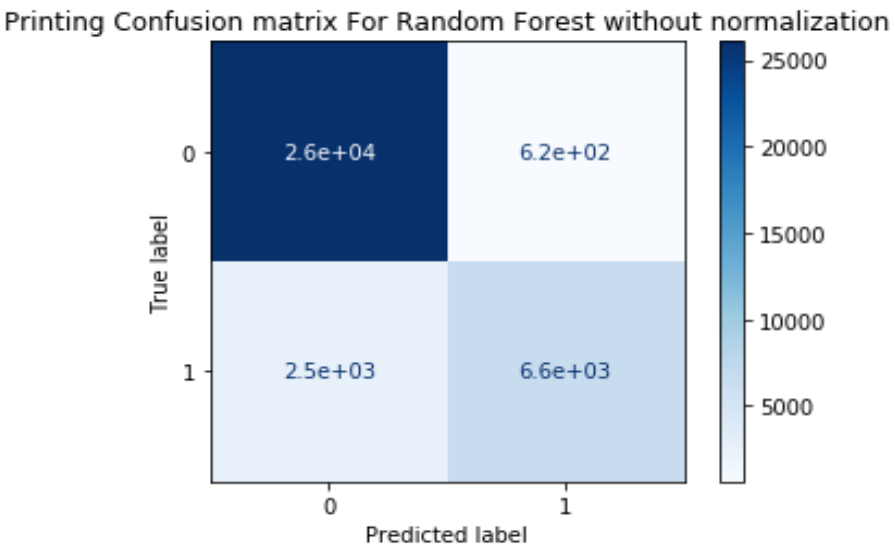


Illustration16 Showing Confusion Matrix without normalization for Random forest

The illustration 16 shows a confusion matrix without normalization for Random Forest using GloVe +TF-IDF for Tweet Level. Each value in the matrix shows the total number of instances in the testing data in the form of an equation. In order to view the exact value of each predicted instance the equation gets converted into numbers printed by the classifier as shown in Table 25

Table 25 Confusion matrix for Random forest

	Depressed	Non-Depressed
Predicted as Depressed	TP =6605	FP =620
Predicted as Non-Depressed	FN =2463	TN =26084

The Table 25 Shows the values of all possible predicted outcomes for Random Forest using GloVe +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 6605 tweets were correctly predicted as depressed by the model

True Negative (TN) = 26084 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 620 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 2463 Tweets were incorrectly predicted as Non-depressed by the model

Table 26 Showing Metric Results for Random forest

Measure	Value
Accuracy	**91.38%**
Recall (Sensitivity)	**72.84%**
Specificity	**97.68%**
Precision	**91.42%**
F1 Score	**94.42%**

Table 26 Shows the results for Random Forest using GloVe +TF-IDF for Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 91.38% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 72.84% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 97.68% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 91.42% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.9442 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

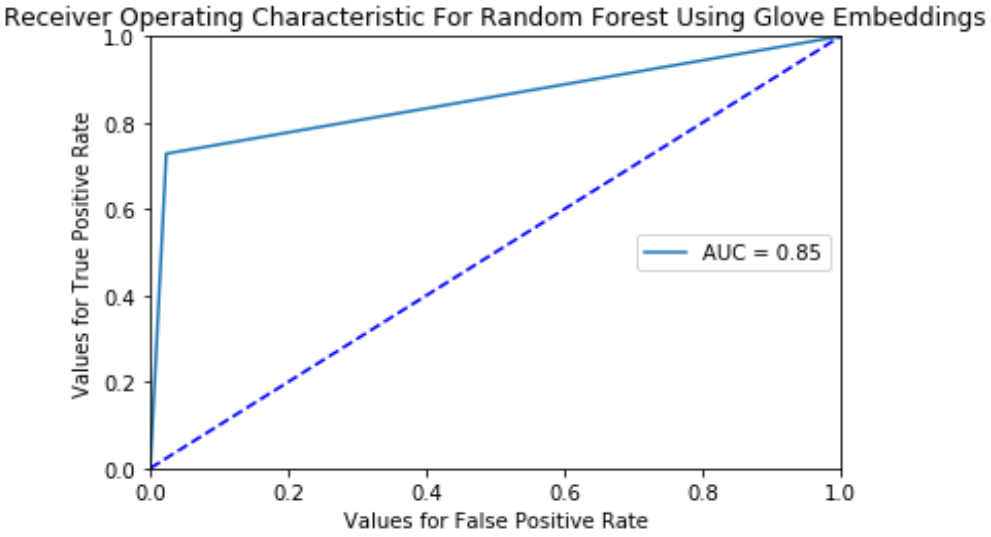


Illustration 17 Showing ROC and AUC for Random forest

The Illustration 17 Shows ROC curve for Random Forest using GloVe +TF-IDF for Tweet Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.85, which lies between the range 0.8 to 0.9 is considered an excellent classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.6.2 Evaluation of Logistic Regression using GloVe +TF-IDF for Tweet Level

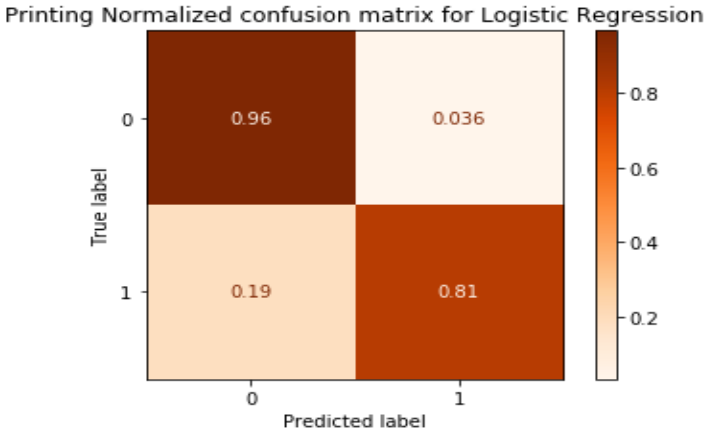


Illustration 18 Showing Normalized Confusion Matrix for Logistic Regression

The Illustration 18 shows a normalized confusion matrix for Logistic Regression using GloVe +TF-IDF for Tweet Level. The decimal representation helps to understand the results by observation of the diagonal values showing the degree of accurately predicted occurrences, which are higher than the off diagonal values. Moreover, the sum of values of each row is equal to 1.00 that shows the 100% utility of possible outcomes for each class.

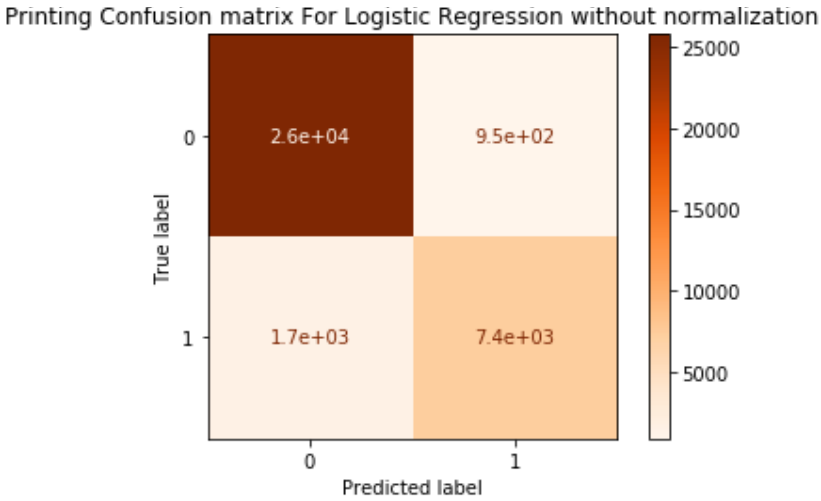


Illustration 19 Showing Confusion Matrix without normalization for Logistic Regression

The illustration 19 shows a confusion matrix without normalization for Logistic Regression using GloVe +TF-IDF for Tweet Level. Each value in the matrix shows the total number of instances in the testing data in the form of an equation. In order to view the exact value of each predicted instance the equation gets converted into numbers printed by the classifier as shown in Table 27

Table 27 Showing Confusion Matrix for Logistic Regression

	Depressed	Non-Depressed
Predicted as Depressed	TP = 7358	FP = 950
Predicted as Non-Depressed	FN =1710	TN = 25754

The Table 27 Shows the values of all possible predicted outcomes for Logistic Regression using GloVe +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 7358 tweets were correctly predicted as depressed by the model

True Negative (TN) = 25754 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 950 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 1710 Tweets were incorrectly predicted as Non-depressed by the model

Table 28 Showing Results for logistic regression

Measure	Value
Accuracy	**92.56%**
Recall(Sensitivity)	**81.14%**
Specificity	**96.44%**
Precision	**88.57%**
F1 Score	**84.69%**

Table 28 Shows the results for Logistic Regression using GloVe +TF-IDF for Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 92.56% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 81.14% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 96.44% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 88.57% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.8469 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

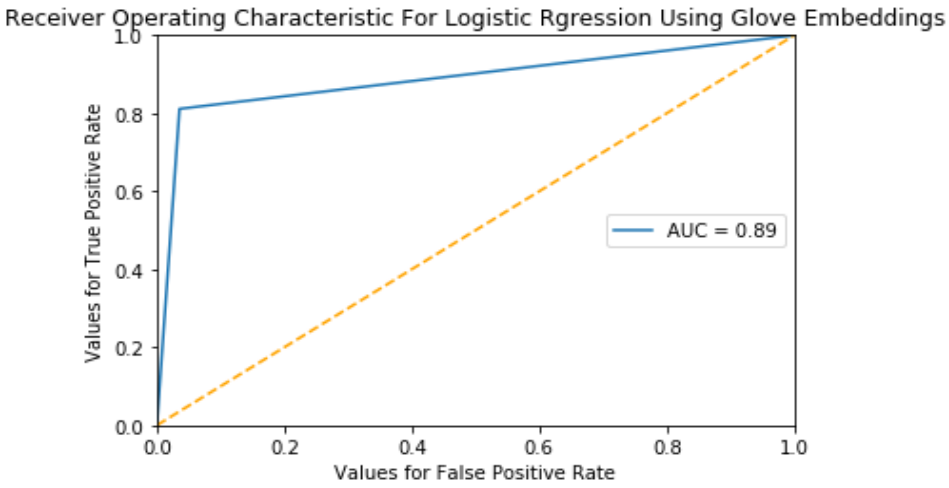


Illustration 20 Showing ROC and AUC for logistic regression

The Illustration 20 Shows ROC curve for Logistic Regression using GloVe +TF-IDF for Tweet Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.89, which lies between the range 0.8 to 0.9 is considered an excellent classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.6.3 Evaluation of SVM using GloVe +TF-IDF For Tweet Level:

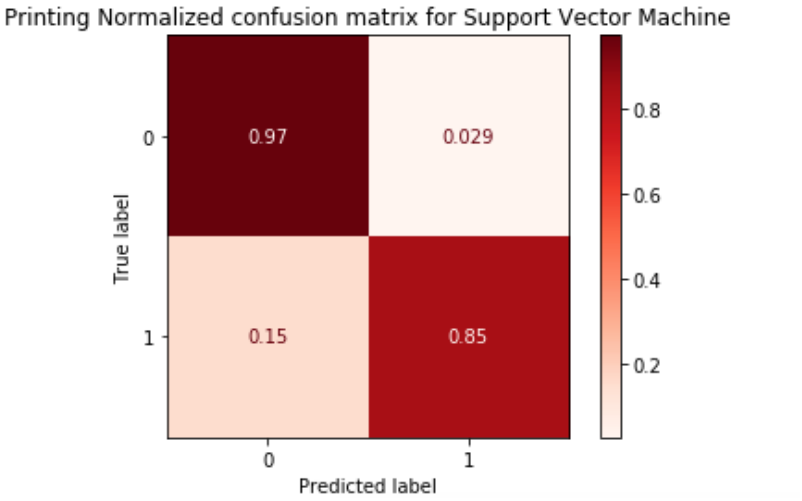


Illustration 21 Showing Normalized Confusion Matrix for SVM

The Illustration 21 shows a normalized confusion matrix for SVM using GloVe +TF-IDF for Tweet Level. The decimal representation helps to understand the results by observation of the diagonal values showing the degree of accurately predicted occurrences, which are higher than the off diagonal values. Moreover, the sum of values of each row is equal to 1.00 that shows the 100% utility of possible outcomes for each class.

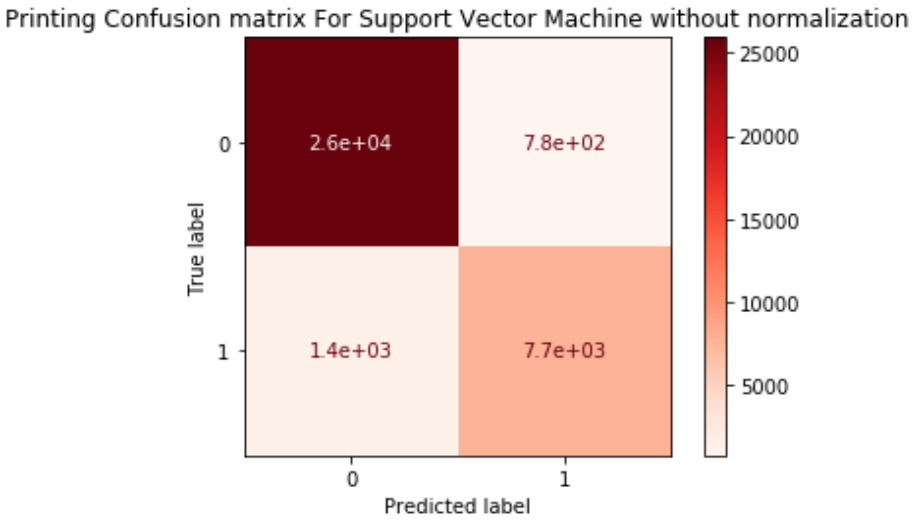


Illustration 22 Showing Confusion Matrix without normalization for SVM

The illustration 22 shows a confusion matrix without normalization for SVM using GloVe +TF-IDF for Tweet Level. Each value in the matrix shows the total number of instances in the testing data in the form of an equation. In order to view the exact value of each predicted instance the equation gets converted into numbers printed by the classifier as shown in Table 29

Table 29 Showing Confusion Matrix for SVM

	Depressed	Non-Depressed
Predicted as Depressed	TP = 7665	FP = 777
Predicted as Non-Depressed	FN = 1403	TN = 25927

The Table 29 Shows the values of all possible predicted outcomes for SVM using GloVe +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 7665 tweets were correctly predicted as depressed by the model

True Negative (TN) = 25927 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 777 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 1403 Tweets were incorrectly predicted as Non-depressed by the model

Table 30 Showing Results for SVM

Measure	Value
Accuracy	**93.91%**
Recall(Sensitivity)	**84.53%**
Specificity	**97.09%**
Precision	**90.80%**
F1 Score	**87.55%**

Table 30 Shows the results for SVM using GloVe +TF-IDF for Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 93.91% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 84.53% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 97.09% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 90.80% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.8755 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

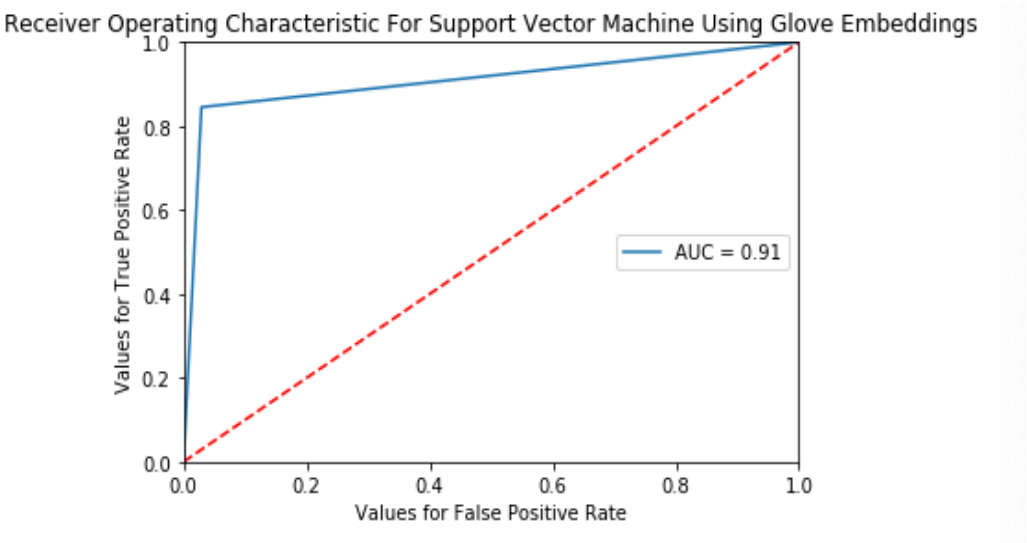


Illustration 23 Showing Roc and AUC for SVM

The Illustration 23 Shows ROC curve for SVM using GloVe +TF-IDF for Tweet Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.91, which is above the range 0.80 to 0.90 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.6.4 Evaluation of Gradient Boosting with Default Decision Trees as weak learner using GloVe +TF-IDF:

Table 31 Confusion matrix for Gradient Boosting

	Depressed	Non-Depressed
Predicted as Depressed	TP = 6933	FP = 820
Predicted as Non-Depressed	FN = 2135	TN = 25884

The Table 31 Shows the values of all possible predicted outcomes for Gradient Boosting with default Decision Trees as weak learner using GloVe +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 6933 tweets were correctly predicted as depressed by the model

True Negative (TN) = 25884 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 820 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 2135 Tweets were incorrectly predicted as Non-depressed by the model

Table 32 Showing Metric Results for Gradient Boosting

Measure	Value
Accuracy	**91.73%**
Recall(Sensitivity)	**77.02%**
Specificity	**96.93%**
Precision	**85.3%**
F1 Score	**82.43%**

Table 32 Shows the results for Gradient Boosting with default Decision Trees as weak learner using GloVe +TF-IDF at Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 92.56% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 81.14% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 96.44% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 88.57% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.8469 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having

balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

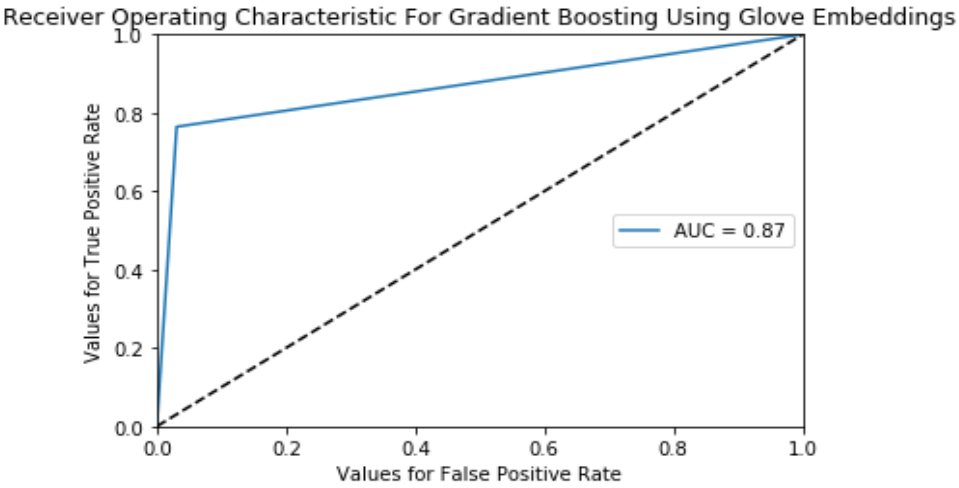


Illustration 24 Showing ROC and AUC for Gradient Boost

The Illustration 24 Shows ROC curve for Gradient Boosting with default Decision Trees as weak learner using GloVe +TF-IDF at Tweet Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.87, which lies between the range 0.8 to 0.9 is considered an excellent classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.6.5 Evaluation of AdaBoost with default Decision Trees as weak learner using GloVe +TF-IDF:

Table 33 Confusion matrix for AdaBoost

	Depressed	Non-Depressed
Predicted as Depressed	TP = 6985	FP = 1197
Predicted as Non-Depressed	FN = 2083	TN = 25550

The Table 33 Shows the values of all possible predicted outcomes for AdaBoost with default Decision Trees as weak learner using GloVe +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 6985 tweets were correctly predicted as depressed by the model

True Negative (TN) = 25550 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 1197 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 2083 Tweets were incorrectly predicted as Non-depressed by the model

Table 34 Showing Results for AdaBoost

Measure	Value
Accuracy	**90.83%**
Recall(Sensitivity)	**77.03%**
Specificity	**95.52%**
Precision	**85.37%**
F1 Score	**80.99%**

Table 34 Shows the results for AdaBoost with default Decision Trees as weak learner using GloVe +TF-IDF at Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 90.83% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 77.03% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 95.52% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 88.37% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.8099 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

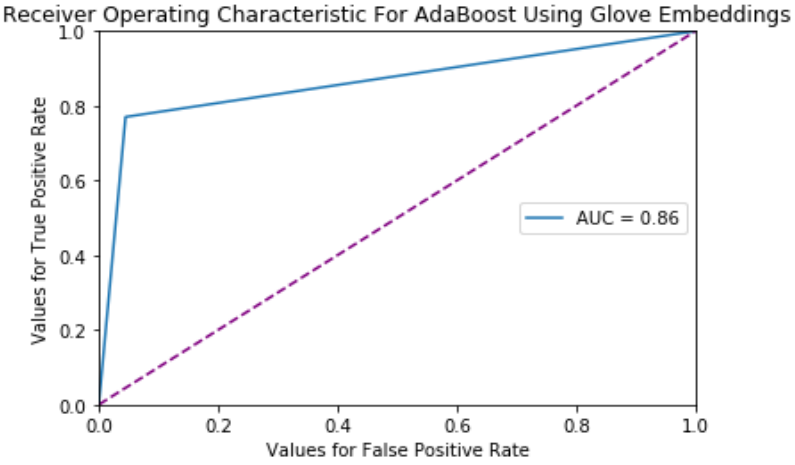


Illustration 25 Showing ROC and AUC for AdaBoost

The Illustration 25 Shows ROC curve for AdaBoost with default Decision Trees as weak learner using GloVe +TF-IDF at Tweet Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.86, which lies between the range 0.8 to 0.9 is considered an excellent classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.6.6 Evaluation of XGBoost with Random Forest as weak learner using GloVe +TF-IDF:

Table 35 Confusion matrix for XGBoost

	Depressed	Non-Depressed
Predicted as Depressed	TP = 6870	FP = 880
Predicted as Non-Depressed	FN = 2012	TN = 26010

The Table 35 Shows the values of all possible predicted outcomes for XGBoost with Random Forest as weak learner using GloVe +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 6870 tweets were correctly predicted as depressed by the model

True Negative (TN) = 26010 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 880 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 2012 Tweets were incorrectly predicted as Non-depressed by the model

Table 36 Showing Results for XGBoost

Measure	Value
Accuracy	**91.91%**
Recall(Sensitivity)	**77.34%**
Specificity	**96.73%**
Precision	**88.64%**
F1 Score	**82.61%**

Table 36 Shows the results for XGBoost with Random Forest as weak learner using GloVe +TF-IDF at Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 91.91% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 77.34% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 96.73% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 88.64% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.8261 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

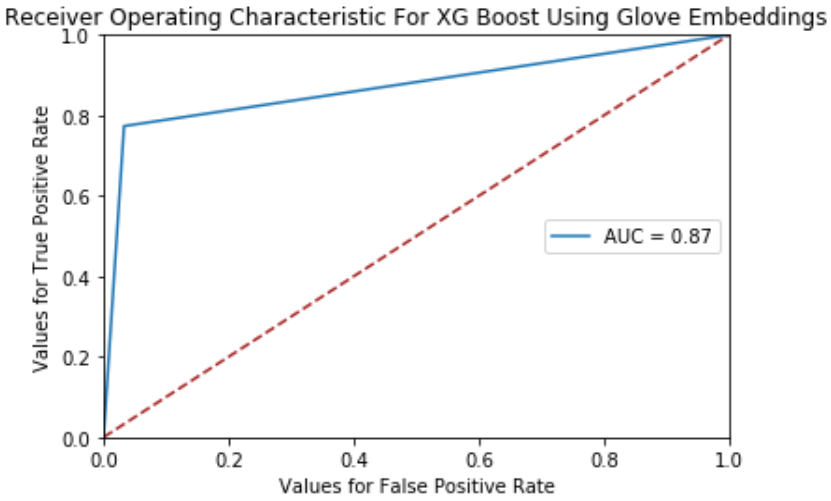


Illustration 26 Showing ROC and AUC For XGBoost

The Illustration 26 Shows ROC curve for XGBoost with Random Forest as weak learner using Word2Vec +TF-IDF at Tweet Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.87, which lies between the range 0.8 to 0.9 is considered an excellent classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.6.7 Evaluation of Long Short-Term Memory (LSTM) for Tweet Level

```
In [172]:
EPOCHS=5 #defining number of epochs
early_stop = EarlyStopping(monitor='val_loss', patience=3) #defining early stop
model_seq_hist = model_lstm_load.fit(x_train, y_train,epochs=EPOCHS, batch_size=12,shuffle=True,callbacks=[early_st

Epoch 1/5
83467/83467 [==============================] - 38s 458us/step - loss: 0.2533 - acc: 0.8888
Epoch 2/5
83467/83467 [==============================] - 39s 467us/step - loss: 0.1989 - acc: 0.9128
Epoch 3/5
83467/83467 [==============================] - 42s 505us/step - loss: 0.1821 - acc: 0.9221
Epoch 4/5
83467/83467 [==============================] - 46s 545us/step - loss: 0.1733 - acc: 0.9246
Epoch 5/5
83467/83467 [==============================] - 38s 450us/step - loss: 0.1681 - acc: 0.9272

In [174]:
labels_pred_seq = model_lstm_load.predict(x_test)
labels_pred_seq = np.round(labels_pred_seq.flatten())
accuracy_seq = accuracy_score(y_test, labels_pred_seq)
print("Accuracy: %.2f%%" % (accuracy_seq*100))

Accuracy: 92.95%

In [175]:
print(classification_report(y_test, labels_pred_seq, digits=3))

              precision    recall  f1-score   support

           0      0.927     0.983     0.954     26702
           1      0.939     0.772     0.847      9070

    accuracy                          0.929     35772
   macro avg      0.933     0.877     0.901     35772
weighted avg      0.930     0.929     0.927     35772
```

Illustration 27 Showing results of LSTM

The Illustration 27 describes the implementation of Long Short-Term Memory (LSTM) on the dataset. The implementation consists of obtaining the word vectors, inside the neural network architecture, and generating padding sequences to prepare the training data. 70 % of the data is split into training set for application into neural networks. Neural networks learn from each loss by correcting their wrong classified mistakes, for which the architecture of epochs has been employed. For training the data, epochs value is set to 5 for this research. Classification report is the metric that inputs the actual and predicted labels and outputs the metrics values of accuracy, recall, precision, f1-score based on their class labels.

Table 37 Showing metrics for Long Short-Term Memory

Measure	Value
Accuracy	**92.95%**
Recall(Sensitivity)	**92.9%**
Precision	**93.0%**
F1 Score	**92.7%**

Table 37 Shows the results for LSTM at Tweet Level in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 92.95% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 92.9% (represents the fraction of tweets detected correctly that actually represents depressed class)

Precision = 93% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.927 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

5.6.8 Evaluation of Bidirectional Long Short-Term Memory (BI-LSTM) at Tweet Level

```
x_train, x_test, y_train, y_test = train_test_split(input_tensor_obtained, all_tweets['target'], test_size=0.3)
```

```
model_lstm.fit(x_train, y_train, batch_size=8, epochs=5) #fitting neural nets on data
```

```
Epoch 1/5
83467/83467 [==============================] - 730s 9ms/step - loss: 0.1761 - accuracy: 0.9319
Epoch 2/5
83467/83467 [==============================] - 732s 9ms/step - loss: 0.1103 - accuracy: 0.9595
Epoch 3/5
83467/83467 [==============================] - 731s 9ms/step - loss: 0.0858 - accuracy: 0.9692
Epoch 4/5
83467/83467 [==============================] - 742s 9ms/step - loss: 0.0672 - accuracy: 0.9765
Epoch 5/5
83467/83467 [==============================] - 746s 9ms/step - loss: 0.0562 - accuracy: 0.9804

<keras.callbacks.callbacks.History at 0x7f9c58058910>
```

```
preds_lstm = model_lstm.predict(x_test)
preds_lstm  = np.round(preds_lstm.flatten())
print(classification_report(y_test, preds_lstm, digits=3))
```

```
              precision    recall  f1-score   support

           0      0.963     0.963     0.963     26702
           1      0.891     0.890     0.891      9070

    accuracy                          0.945     35772
   macro avg      0.927     0.927     0.927     35772
weighted avg      0.945     0.945     0.945     35772
```

Illustration 28 Implementation Results for Bi-LSTM

The Illustration 28 describes the implementation of Bidirectional Long Short-Term Memory (BI-LSTM) on the dataset. The implementation consists of obtaining the word vectors, inside the neural network architecture, and generating padding sequences, according to vocabulary count to prepare the training data. 70 % of the data is split into training set for application into Bi-LSTM. For training the data, epochs value is set to 5 in this research. Classification report is the metric that inputs the actual and predicted labels and outputs the metrics values of accuracy, recall, precision, Fl-score based on their class labels.

Table 38 Showing Results for Bi-LSTM

Measure	Value
Accuracy	**94.5%**
Recall(Sensitivity)	**94.5%**
Precision	**94.5%**
F1 Score	**94.5%**

Table 38 Shows the results for Bi-LSTM at Tweet Level in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 94.5% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 94.5% (represents the fraction of tweets detected correctly that actually represents depressed class)

Precision = 94.5% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.945 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

5.7 Results for Using Trained Word2Vec Embeddings at Tweet Level

This section shows the results of all the advanced machine and deep learning algorithms by using ensemble learning methods combined with Word2Vec and TF-IDF.

5.7.1 Evaluation of Random Forest using Word2Vec +TF-IDF at Tweet Level

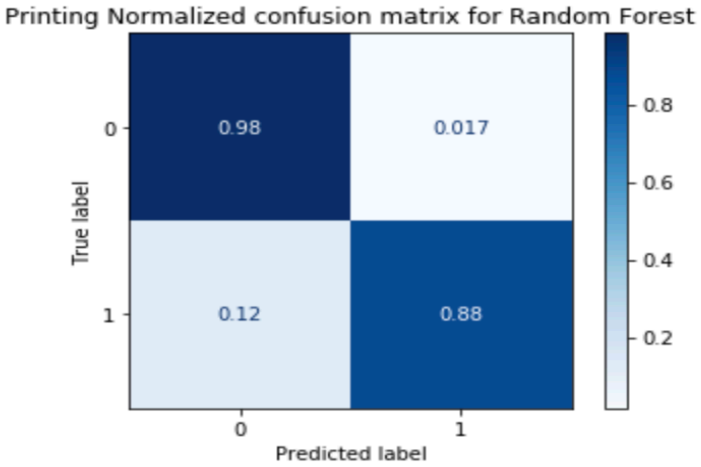


Illustration 29 Normalized Confusion Matrix for Random forest

The Illustration 29 shows a normalized confusion matrix for Random Forest using Word2Vec +TF-IDF for Tweet Level. The decimal representation helps to understand the results by observation of the diagonal values showing the degree of accurately predicted occurrences, which are higher than the off diagonal values. Moreover, the sum of values of each row is equal to 1.00 that shows the 100% utility of possible outcomes for each class.

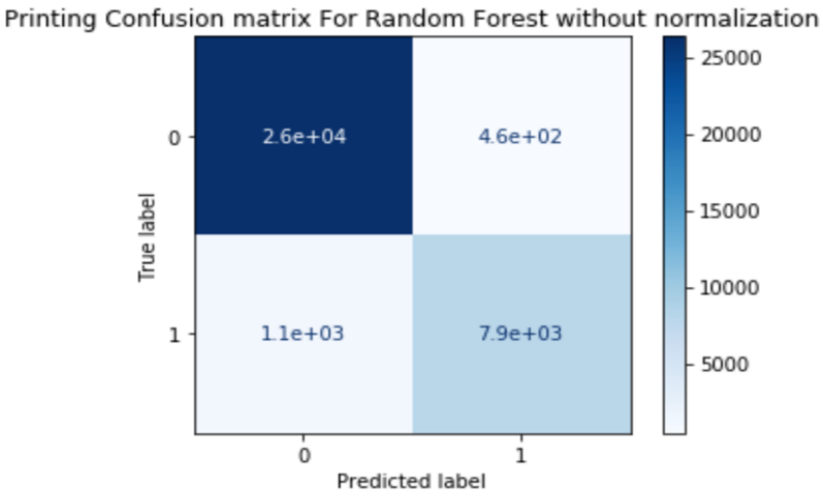


Illustration 30 Showing Confusion Matrix without normalization for Random forest

The illustration 30 shows a confusion matrix without normalization for Random Forest using Word2Vec +TF-IDF for Tweet Level. Each value in the matrix shows the total number of instances in the testing data in the form of an equation. In order to view the exact value of each predicted instance the equation gets converted into numbers printed by the classifier as shown in Table 39

Table 39 Showing Confusion Matrix for Random Forest with Word2Vec and TF-IDF

	Depressed	Non-Depressed
Predicted as Depressed	TP = 7883	FP = 459
Predicted as Non-Depressed	FN = 1094	TN = 26336

The Table 39 Shows the values of all possible predicted outcomes for Random Forest using Word2Vec +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 7883 tweets were correctly predicted as depressed by the model

True Negative (TN) = 26336 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 459 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 1094 Tweets were incorrectly predicted as Non-depressed by the model

Table 40 Showing Results for Random Forest

Measure	Value
Accuracy	**95.65%**
Recall (Sensitivity)	**87.81%**
Specificity	**98.29%**
Precision	**94.49%**
F1 Score	**91.03%**

Table 40 Shows the results for Random Forest using Word2Vec +TF-IDF for Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 95.65% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 87.81% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 98.29% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 94.49% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.9103 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

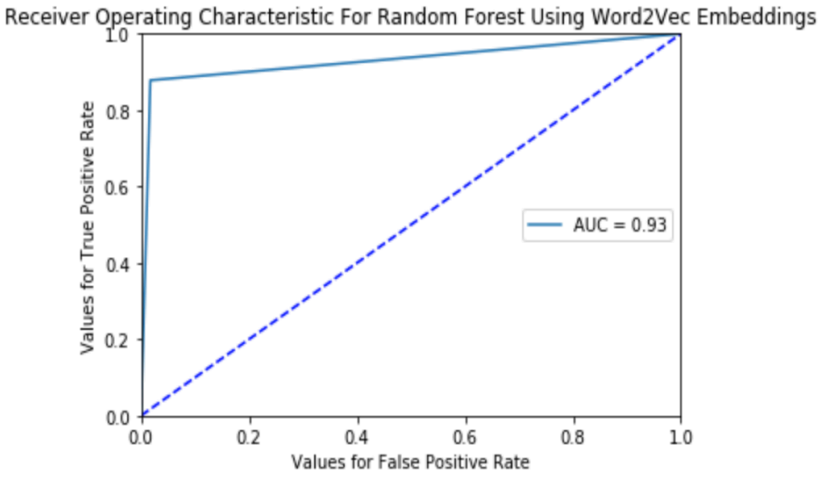


Illustration 31 Showing ROC for Random forest

The Illustration 31 Shows ROC curve for random forest using Word2Vec +TF-IDF for Tweet Level depicting the probabilities of correctly classified values and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.93, which is above the range 0.8 to 0.9 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.7.2 Evaluation of SVM using Word2Vec +TF-IDF at Tweet Level:

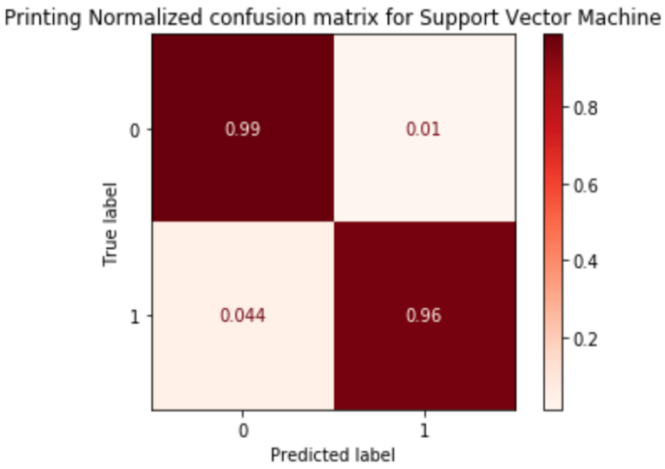


Illustration 32 Showing Normalized Confusion Matrix for SVM

The Illustration 32 shows a normalized confusion matrix for SVM using Word2Vec +TF-IDF for Tweet Level. The decimal representation helps to understand the results by observation of the diagonal values showing the degree of accurately predicted occurrences, which are higher than the off diagonal values. Moreover, the sum of values of each row is equal to 1.00 that shows the 100% utility of possible outcomes for each class.

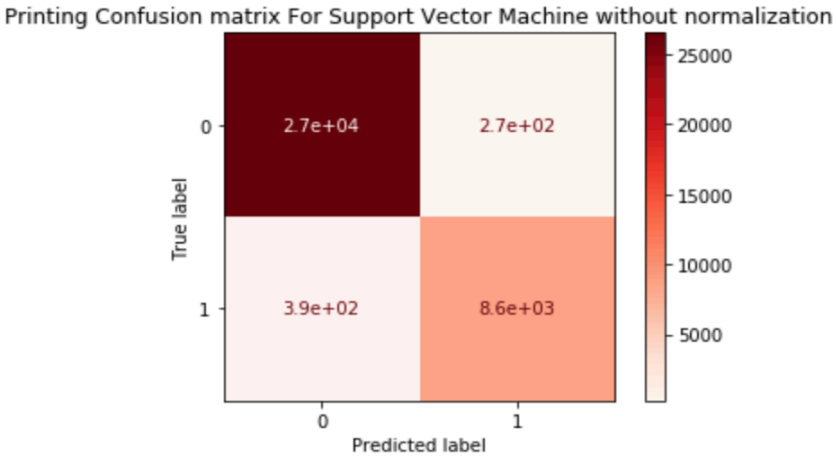


Illustration 33 Showing Confusion Matrix without normalization for SVM

The illustration 33 shows a confusion matrix without normalization for SVM using Word2Vec +TF-IDF for Tweet Level. Each value in the matrix shows the total number of instances in the testing data in the form of an equation. In order to view the exact value of each predicted instance the equation gets converted into numbers printed by the classifier as shown in Table 41

Table 41 Showing Confusion Matrix for SVM using Word2Vec and TF-IDF

	Depressed	Non-Depressed
Predicted as Depressed	TP = 8585	FP = 272
Predicted as Non-Depressed	FN = 392	TN = 26225

The Table 41 Shows the values of all possible predicted outcomes for SVM using Word2Vec +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 8585 tweets were correctly predicted as depressed by the model

True Negative (TN) = 26225 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 272 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 392 Tweets were incorrectly predicted as Non-depressed by the model

Table 42 Showing Results for SVM

Measure	**Value**
Accuracy	**98.14%**
Recall (Sensitivity)	**95.63%**
Specificity	**98.98%**
Precision	**96.93%**
F1 Score	**96.28%**

Table 42 Shows the results for SVM using Word2Vec +TF-IDF for Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 98.14% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 95.63% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 98.98% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 96.93% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.9628 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

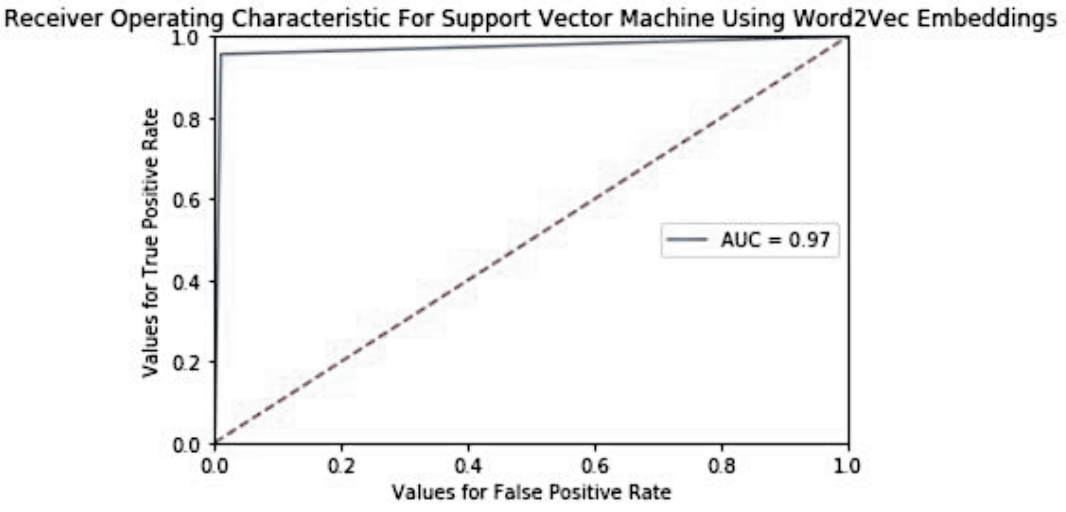


Illustration 34 Showing ROC and AUC for SVM

The Illustration 34 Shows ROC curve for SVM using Word2Vec +TF-IDF for Tweet Level depicting the probabilities of correctly classified values and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.97, which is above the range 0.8 to 0.9 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.7.3 Evaluation of Logistic Regression using Word2Vec +TF-IDF:

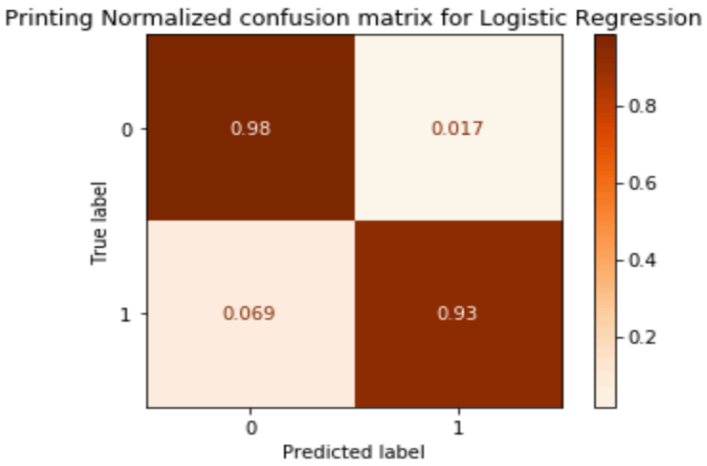


Illustration 35 Showing Normalized Confusion Matrix for Logistic Regression

The Illustration 35 shows a normalized confusion matrix for Logistic Regression using Word2Vec +TF-IDF for Tweet Level. The decimal representation helps to understand the results by observation of the diagonal values showing the degree of accurately predicted occurrences, which are higher than the off diagonal values. Moreover, the sum of values of each row is equal to 1.00 that shows the 100% utility of possible outcomes for each class.

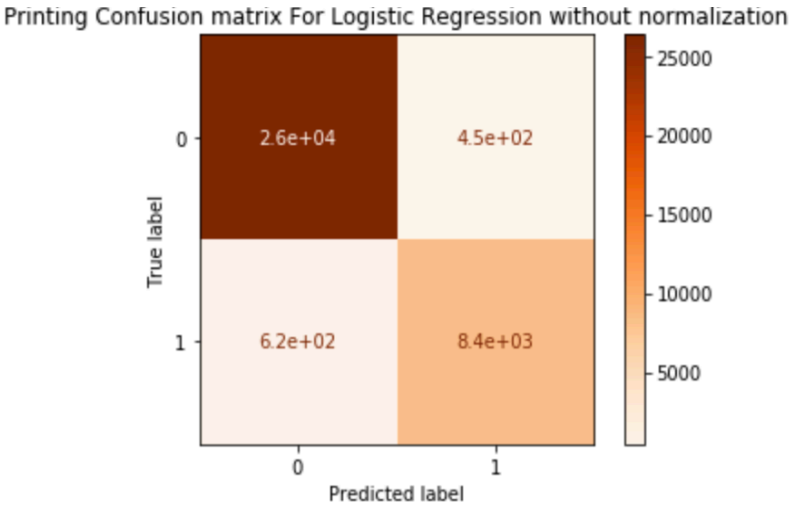


Illustration 36 Showing Confusion Matrix without normalization for Logistic Regression

The illustration 36 shows a confusion matrix without normalization for Logistic Regression using Word2Vec +TF-IDF for Tweet Level. Each value in the matrix shows the total number of instances in the testing data in the form of an equation. In order to view the exact value of each predicted instance the equation gets converted into numbers printed by the classifier as shown in Table 43

Table 43 Showing Confusion Matrix for Logistic Regression

	Depressed	Non-Depressed
Predicted as Depressed	TP = 8362	FP = 446
Predicted as Non-Depressed	FN = 615	TN = 26349

The Table 43 Shows the values of all possible predicted outcomes for Logistic Regression using Word2Vec +TF-IDF at Tweet Level. The values in the matrix are as follows:

True Positive (TP) = 8362 tweets were correctly predicted as depressed by the model

True Negative (TN) = 26349 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 446 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 615 Tweets were incorrectly predicted as Non-depressed by the model

Table 44 Showing Results for Logistic Regression

Measure	Value
Accuracy	**97.03%**
Recall(Sensitivity)	**93.15%**
Specificity	**98.34%**
Precision	**94.94%**
F1 Score	**94.03%**

Table 44 Shows the results for Logistic Regression using Word2Vec +TF-IDF for Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 97.03% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 93.15% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 98.34% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 94.94% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.9403 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

Illustration 37 Showing ROC and AUC for Logistic Regression

The Illustration 37 Shows ROC curve for Logistic Regression using Word2Vec +TF-IDF at Tweet Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.96, which is above the range 0.8 to 0.9 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.7.4 Evaluation of AdaBoost with default decision trees as weak learner using Word2Vec +TF-IDF

Table 45 Confusion Matrix Results for AdaBoost

	Depressed	Non-Depressed
Predicted as Depressed	TP = 7828	FP = 790
Predicted as Non-Depressed	FN = 1149	TN = 26005

The Table 45 Shows the values of all possible predicted outcomes for AdaBoost with default decision trees as weak learner using Word2Vec +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 7828 tweets were correctly predicted as depressed by the model

True Negative (TN) = 26005 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 790 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 1149 Tweets were incorrectly predicted as Non-depressed by the model

Table 46 Showing Metric Results for AdaBoost

Measure	Value
Accuracy	**94.58%**
Recall(Sensitivity)	**87.20%**
Specificity	**97.05%**
Precision	**90.83%**
F1 Score	**88.98%**

Table 46 Shows the results for AdaBoost with default decision trees as weak learner using Word2Vec +TF-IDF for Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 94.58% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 87.20% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 97.05% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 90.83% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.8898 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

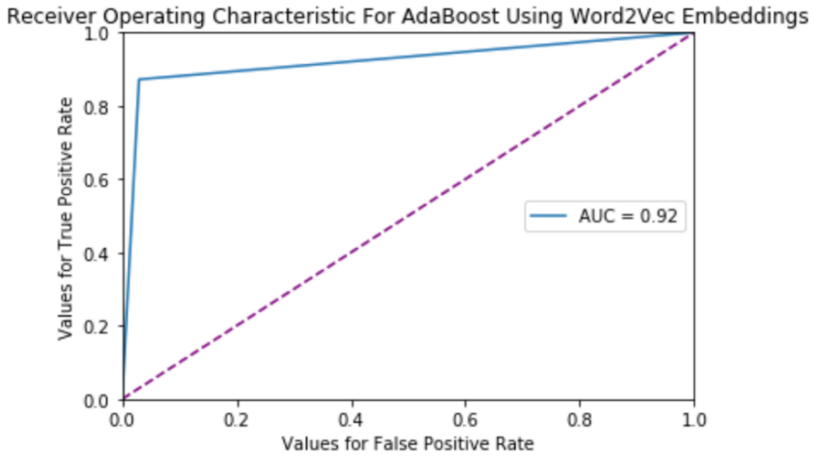


Illustration 38 Showing ROC and AUC for AdaBoost

The Illustration 38 Shows ROC curve for AdaBoost with default decision trees as weak learner using Word2Vec +TF-IDF at Tweet Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.92, which is above the range 0.8 to 0.9 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.s

5.7.5 Evaluation of Gradient Boosting with default decision trees as weak learner using Word2Vec +TF-IDF

Table 47 Showing Confusion Matrix for Gradient Boosting Classifier

	Depressed	Non-Depressed
Predicted as Depressed	TP = 7927	FP = 558
Predicted as Non-Depressed	FN = 1050	TN = 26237

The Table 47 Shows the values of all possible predicted outcomes for Gradient Boosting with default decision trees as weak learner using Word2Vec +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 7927 tweets were correctly predicted as depressed by the model

True Negative (TN) = 26237 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 558 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 1050 Tweets were incorrectly predicted as Non-depressed by the model

Table 48 Showing Metrics Results for Gradient Boosting

Measure	**Value**
Accuracy	**95.50%**
Recall(Sensitivity)	**87.20%**
Specificity	**97.92%**
Precision	**90.83%**
F1 Score	**90.79%**

Table 48 Shows the results for Gradient Boosting with default decision trees as weak learner using Word2Vec +TF-IDF for Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 95.50% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 87.20% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 97.92% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 90.83% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.9079 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

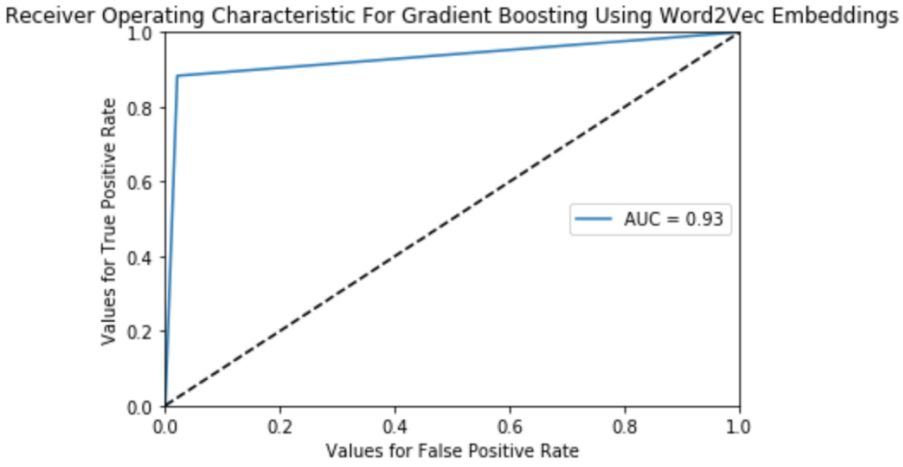


Illustration 39 Showing ROC and AUC for Gradient Boost

The Illustration 39 Shows ROC curve for Gradient Boosting with default decision trees as weak learner using Word2Vec +TF-IDF at Tweet Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.93, which is above the range 0.8 to 0.9 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.7.6 Evaluation of XGBoost with Random Forest as weak learner using Word2Vec +TF-IDF

Table 49 Showing Confusion Matrix Results for XGBoost

	Depressed	Non-Depressed
Predicted as Depressed	TP = 7992	FP = 570
Predicted as Non-Depressed	FN = 985	TN = 26225

The Table 49 Shows the values of all possible predicted outcomes for XGBoost with Random Forest as weak learner using Word2Vec +TF-IDF at Tweet Level. The values in the matrix are explained as follows:

True Positive (TP) = 7358 tweets were correctly predicted as depressed by the model

True Negative (TN) = 25754 tweets were correctly predicted as Non-depressed by the model

False Positive (FP) = 950 Tweets were incorrectly predicted as Depressed by the model

False Negative (FN) = 1710 Tweets were incorrectly predicted as Non-depressed by the Model

Table 50 Showing Results for XGBoost

Measure	Value
Accuracy	**95.65%**
Recall(Sensitivity)	**89.03%**
Specificity	**97.87%**
Precision	**93.34%**
F1 Score	**91.13%**

Table 50 Shows the results for XGBoost with Random Forest as weak learner using Word2Vec +TF-IDF at Tweet Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Accuracy = 95.656% (represents the total accuracy of prediction performance by the model)

Recall (Sensitivity) = 89.03% (represents the fraction of tweets detected correctly that actually represents depressed class)

Specificity = 97.87% (represents the fraction of tweets detected correctly that actually represents non-depressed class)

Precision = 93.34% (represents the fraction of possible predicted outcomes that were significant)

F1 score in decimal value = 0.9113 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

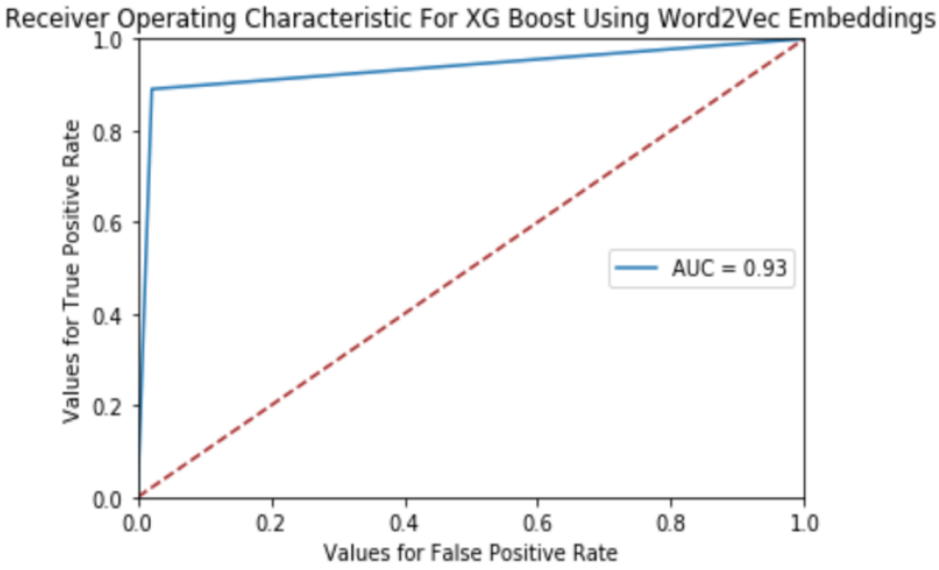


Illustration 40 Showing ROC and AUC for XGBoost

The Illustration 40 Shows ROC curve for XGBoost with Random Forest as weak learner using Word2Vec +TF-IDF for Tweet Level, depicting the probabilities of correctly classified values and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.93, which is above the range 0.8 to 0.9 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

5.8 Comparison of All Models Under Glove + TF-IDF and Word2Vec + TF-IDF for Tweet to Tweet Level

Table 51 showing Comparison of All Models Under Tweet Level

ALGORITHMS	ACCURACY OF MODELS WITH GLOVE + TF-IDF	ACCURACY OF MODELS WITH WORD2VEC + TF-IDF
RANDOM FOREST	91.38%	95.65%
SVM	93.91%	98.14%
LOGISTIC REGRESSION	92.56%	97.03%
XGBOOST	91.91%	95.65%
ADABOOST	90.83%	94.58%
GRADIENT BOOST	91.73%	95.50%

Table 51 represents the comparison of accuracies for all algorithms under Glove + TF-IDF and Word2Vec + TF-IDF at Tweet Level. The highest accuracy is achieved by SVM model with and Word2Vec + TF-IDF recorded as 98.14% followed by the second highest accuracy of 97.03% under Logistic Regression with Word2Vec + TF-IDF.

Table 52 showing Comparison of LSTM and Bi-LSTM Under Experiments for Tweet Level

ALGORITHMS	METRICS				
	PRECISION	RECALL	F1-SCORE	ACCURACY	TIME COMPLEXITY
LSTM	93.0%	92.9%	92.7%	92.9%	38 sec
BI-LSTM	94.5%	94.5%	94.5%	94.5%	746 sec

Table 52 shows the accuracies of LSTM and BI-LSTM using various metrics where BI-LSTM scores the highest accuracy of 94.5% with a recall value of 94.5%. However, the time complexity of LSTM model is faster than the run time complexity of BI-LSTM.

5.9 Comparison of All Models under Glove + TF-IDF and Word2Vec + TF-IDF at Tweet Level Using Various Metrics

Table 53 showing Comparison of All Models using various metrics under tweet level

ALGORITHMS	TECHNIQUES	METRICS			
		PRECISION	RECALL	F1-SCORE	ACCURACY
RANDOM FOREST	GLOVE + TF-IDF	91.42%	72.84%	94.42%	91.38%
	WORD2VEC + TF-IDF	94.49%	87.81%	91.03%	95.65%
SVM	GLOVE + TF-IDF	90.80%	84.53%	87.55%	93.91%
	WORD2VEC + TF-IDF	**96.73%**	**95.63%**	**96.28%**	**98.14%**
LOGISTIC REGRESSION	GLOVE + TF-IDF	88.57%	81.14%	84.69%	92.56%
	WORD2VEC + TF-IDF	**94.94%**	**93.15%**	**94.03%**	**97.03%**
XGBOOST	GLOVE + TF-IDF	88.64%	77.34%	82.61%	91.91%
	WORD2VEC + TF-IDF	93.34%	89.03%	91.13%	95.65%
GRADIENT BOOST	GLOVE + TF-IDF	85.31%	77.02%	82.43%	91.73%
	WORD2VEC + TF-IDF	90.83%	87.20%	90.79%	95.50%
ADABOOST	GLOVE + TF-IDF	85.37%	77.03%	80.99%	90.83%
	WORD2VEC + TF-IDF	90.83%	87.20%	88.98%	94.58%

Table 53 shows comparison of all algorithms under Glove + TF-IDF and Word2Vec + TF-IDF at Tweet Level using various metrics such as Precision, Recall, F1-score and accuracy. However, recall is the most important score because it tells the prediction rate for true positive and true negative values that are correctly classified. The highest accuracy is recorded at 98.14% with a recall rate of 95.63% by the SVM algorithm with Word2Vec + TF-IDF. However, the second highest scores recorded at 97.03% accuracy and 93.15% recall are shown by Logistic regression with Word2vec + TF-IDF.

5.10 Comparison of All Models Under Glove + TF-IDF and Word2Vec + TF-IDF at Tweet Level Using K-Fold Cross Validation Scores

Table 54 showing Comparison using K-fold on all models under Tweet Level

K-FOLD CROSS VALIDATION SCORES (K=10)		
ALGORITHMS	10-CV SCORES OF MODELS WITH GLOVE + TF-IDF	10-CV SCORES OF MODELS WITH WORD2VEC + TF-IDF
RANDOM FOREST	90.94%	95.41%
SVM	**93.41%**	**97.80%**
LOGISTIC REGRESSION	**92.49%**	**96.97%**
XGBOOST	65.28%	79.80%
ADABOOST	90.62%	94.55%
GRADIENT BOOST	91.72%	95.33%

The table 54 shows Comparison of all models with Under Glove + TF-IDF and Word2Vec + TF-IDF at Tweet Level using k-fold cross validation scores. The cross validation using K-folds helps to verify the model by testing various samples for producing an average outcome in terms of accuracy. The K in the table shows the folds holding different section of testing samples. The highest K-fold cross validation scores are recorded at 97.80% by SVM with Word2Vec + TF-IDF.

5.11 Roc Curve for GloVe + TF-IDF and Word2Vec + TF-IDF

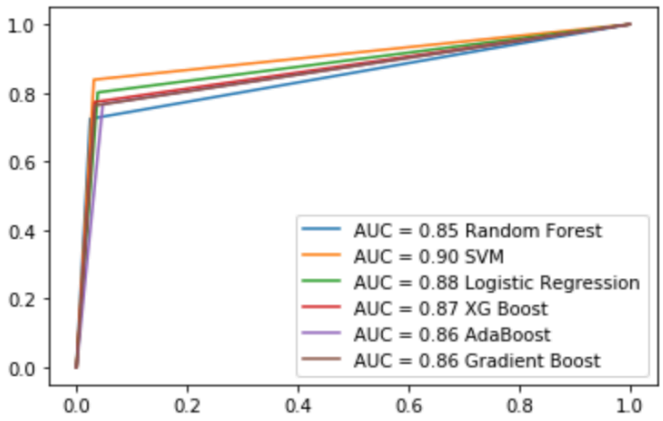


Illustration 41 Comparing ROC and AUC for different models for GloVe and TF-IDF

Figure (41), depicts comparison of ROC curves of all the models built on the top of the word embeddings from GloVe and TF-IDF. The model with the best AUC value is SVM, with a value of 0.90 whereas Random Forest AUC value is least among all models that is 0.85.

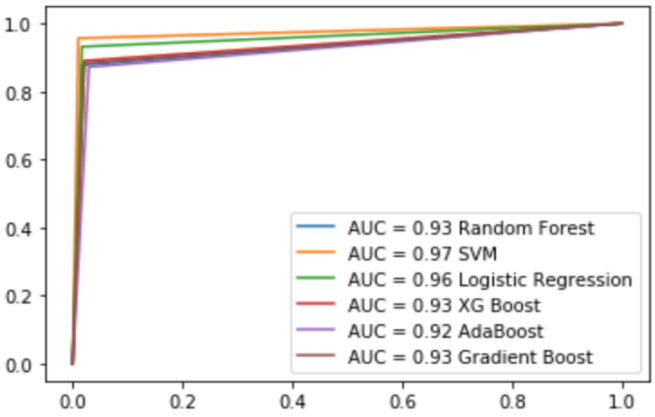


Illustration 42 Comparison of ROC and AUC value among all models with Word2Vec and TF-IDF

Figure (42) depicts comparison of ROC curves of all the models built on the top of the word embeddings from Word2Vec and TF-IDF. The model with the best AUC value is SVM, with a value of 0.97 whereas AdaBoost AUC value is least among all models that is 0.92.

5.12 Visualization of Data – Further Insights

Apart from the experimental results, there are many factors that can be utilized from user's timeline that equally contribute to the detection of depression. These factors help to solve various information under the hidden layers of tweets including the co-relation of the elements such as Time, Age, Location, Intensity of Words as well as other factors discussed in the following topics.

5.12.1 Word Cloud for Depressive and Non-Depressive Tweet

According to [1], the language used by emotionally distressed people and non-depressed people varies in terms of usage of words. The depressed user while expressing themselves are more inclined towards usage of words that reflect negative tone and emotion. By using such vocabulary, they believe that they can be noticed by others in the same community or perhaps, can be helped by someone, for fighting the difficulty. The two figures 43 and 44, (Word clouds with top 100 terms) illustrates the contrast of the language used by depressed and non-depressed users respectively. First word cloud shows the negative polarity words such as fighting, anxiety, depressed, antidepressant, whereas the other word cloud contains words such as amazingly, happy, love, friend.

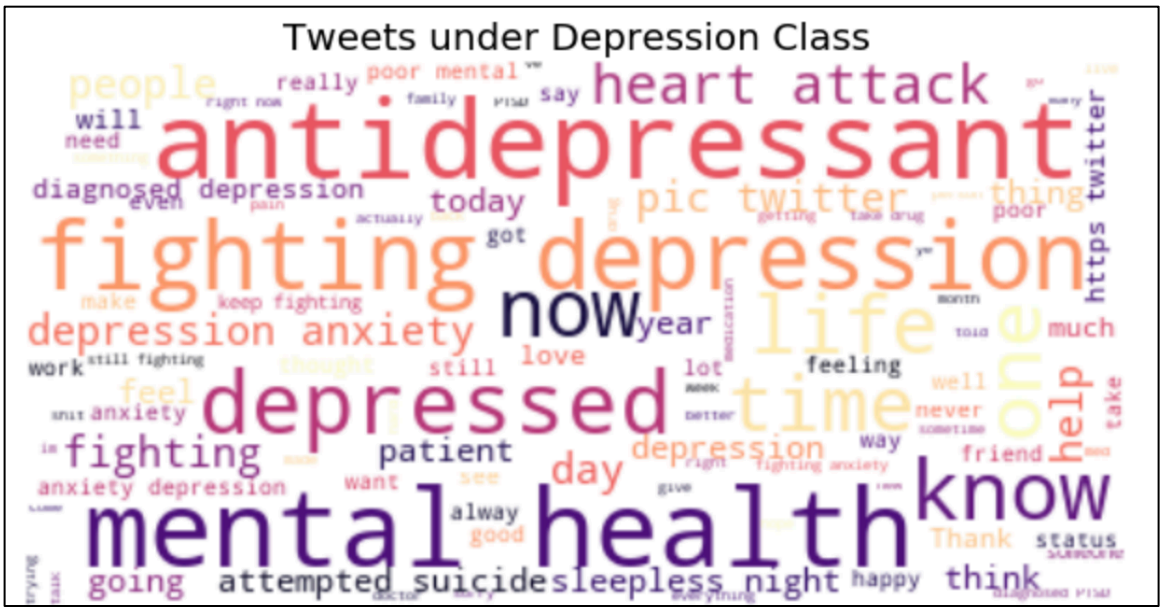


Illustration 43 Showing Word cloud for depressive class

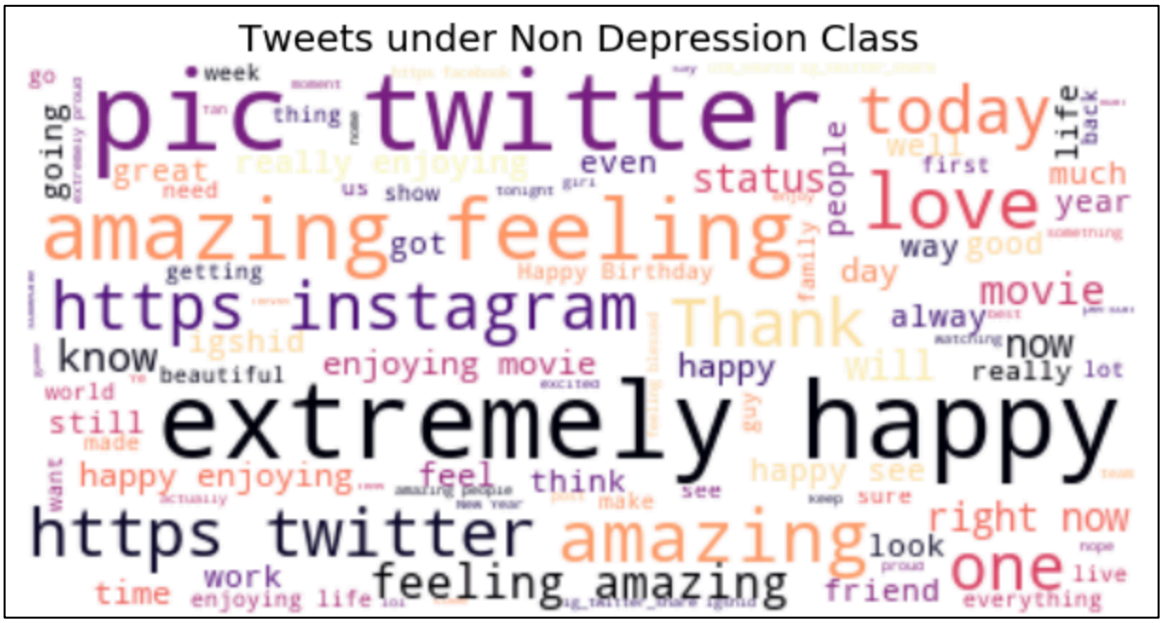


Illustration 44 Showing Word cloud for Non-depressive class

The Illustration 43 shows the top 100 terms used by users under the depressive class. On the other hand, the Illustration 44 shows the top 100 terms used by the users under the Non-depressive class.

5.12.2 Visualization for location of users

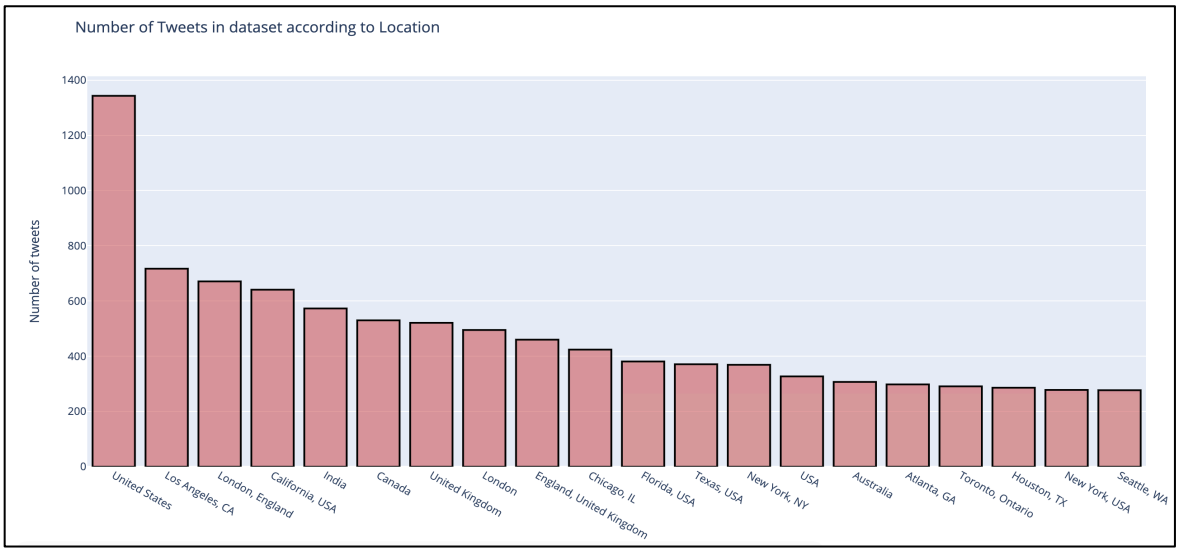


Illustration 45 Showing Location statistics

The illustration 45 shows the tweets collected by the Users for the creation of the dataset for this research also facilitates with the location parameter. The graph above describes the locations of different users whose tweets were downloaded and scrutinized. The maximum number of users present in the dataset belong to United States. The other dominant countries in this dataset include United Kingdom, India, Russia, Canada, Australia.

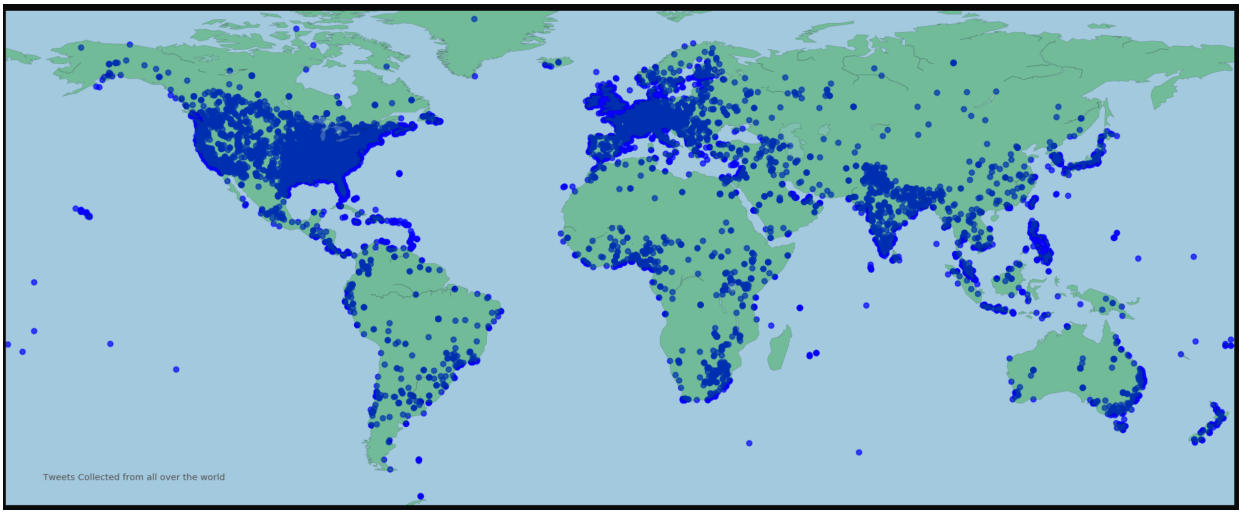


Illustration 46 Showing Location statistics for Total tweets

The figure 46 shows the dispersion or scattering of data points in the form of total collected data containing depressive and non-depressive tweets represented as blue colour from all over the world providing the detailed scenario of the data in a compact view. The data points replicate the trend as shown in the bar graph (figure 39). Further, the country showing maximum number of tweets is United States and the other major countries include India and United Kingdom

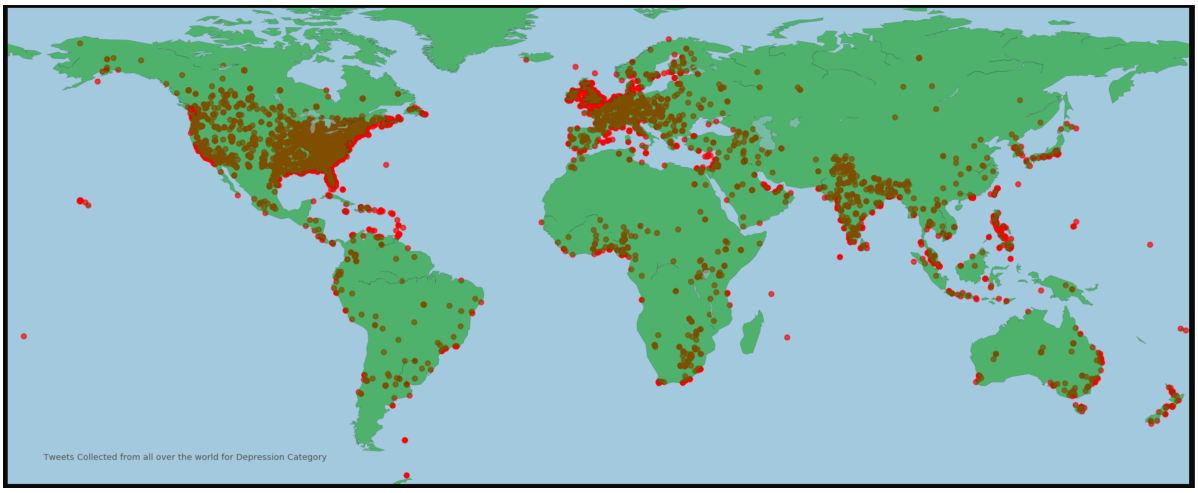


Illustration 47 Showing statistics of locations for depressed tweets

The figure 47 illustrates the spread of depressive tweets in red colour belonging to the depressed category. This picture depicts that the spread of depression candidates is widespread among almost all the countries, which are dominant in the dataset. This trend is observed in all parts of the world that are hit by depression.

5.12.3 Distribution of Hours based on category of Depressed/Non-depressed class

Table 55 Showing hour distribution

Hour	Depressed	Non-depressed
0	1237	4994
1	1232	4535
2	1377	4303
3	1564	4377
4	1536	3679
5	1522	3034
6	1440	2543
7	1244	2380
8	1049	2225
9	938	1982
10	829	1746
11	782	1715
12	664	1948
13	817	2302
14	962	2810
15	1176	3442
16	1192	3707
17	1297	3759
18	1562	4214
19	1504	4320
20	1768	4556
21	1436	4998
22	1236	5586
23	1235	4991

The studies reveal that hour of the day also plays a vital role in determining the emotional state of the person. It can be observed from the table 55, that the depressed people tend to use social media during late night hours of the day. The data frame describes the distribution of tweets among depressed and non-depressed category between different

hours of the day. While the maximum number of depressed users tweeted at 8 pm of the day, while mostly non-depressed users tweeted at 10 pm. There are also emerging facts that can be analyzed by distribution of this chart. Around 5% of people belonging to depressed category, were active at 3 am of the day, whereas when the same number of samples of non-depressed users is taken into consideration, then this ratio is reduced to roughly 1.5% of the non-depressed population.

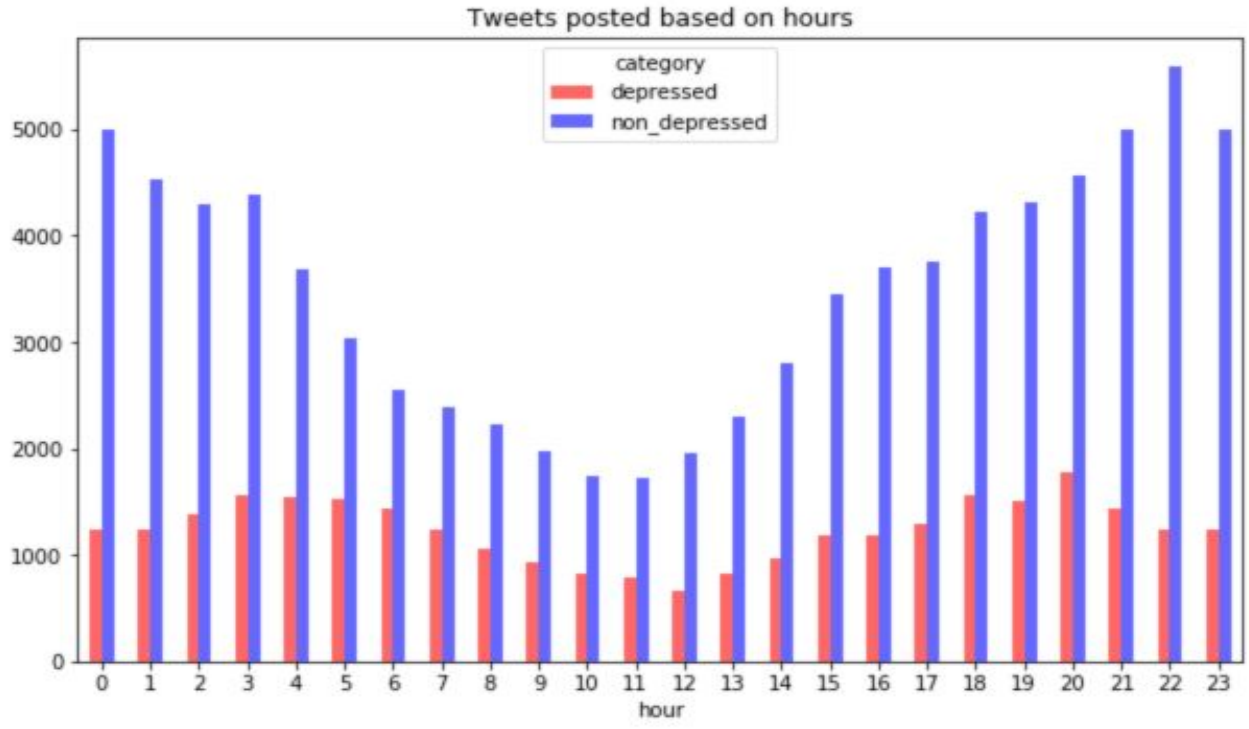


Illustration 48 Showing hour-based tweets of depressed and non-depressed category

The bar graph (illustration 48) is the pictorial representation of the data frame, imitating the proven results, describing the distribution of hours of posting among depressed and non-depressed users. The users of both classes were less active during the mid-hours of the day predicting general tendency of the usage of social media whereas the number of depressed users were more active during mid-morning hours from 3 am to 6 am.

Table 56 Percentage statistics by hour

Percentage Split of Hours		
Hour	Depressed	Non-depressed
0	4.179195	5.934923
1	4.162303	5.389442
2	4.652184	5.113731
3	5.283962	5.201673
4	5.189365	4.372163
5	5.142066	3.605638
6	4.865029	3.022128
7	4.202845	2.828417
8	3.544039	2.644214
9	3.169026	2.355430
10	2.800770	2.074965
11	2.641981	2.038124
12	2.243319	2.315024
13	2.760228	2.735721
14	3.250110	3.339434
15	3.973107	4.090509
16	4.027163	4.405438
17	4.381905	4.467236
18	5.277205	5.007962
19	5.081253	5.133934
20	5.973175	5.414399
21	4.851515	5.939676
22	4.175817	6.638462
23	4.172438	5.93135

The table 56 shows the Percentage spilt of hours for depressed and non-depressed people spending time on social media in 0 to 23 hours. For e.g. 4.17 % of depressed people spend time on social media in 0 hour . Similarly, 5.93 % non-depressed people spend time on social media in 0 hour.

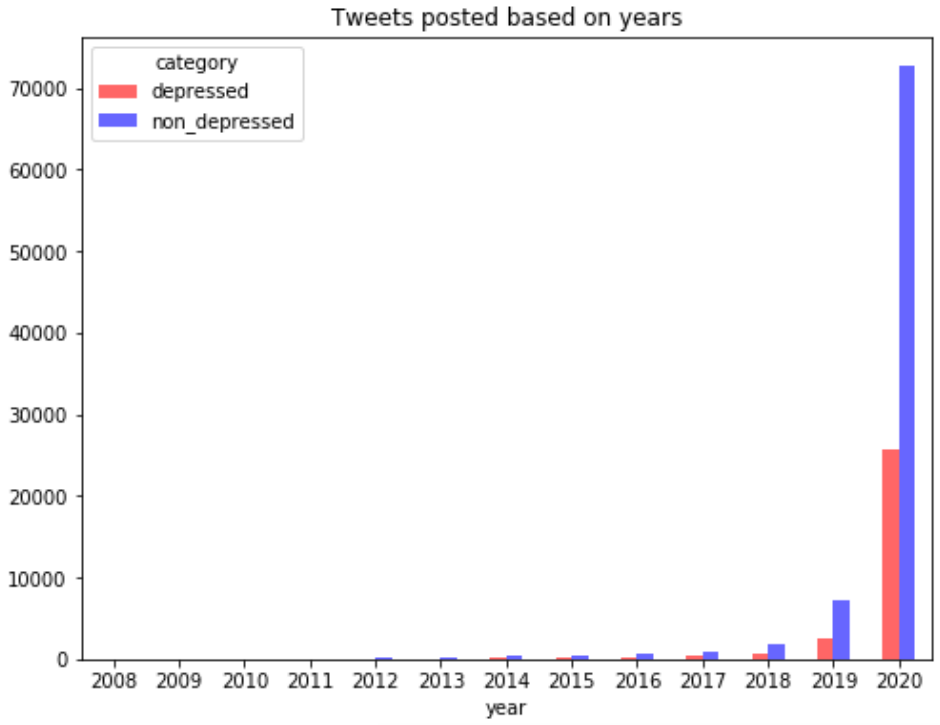


Illustration 49 Showing pattern of users during years

The bar graph (illustration 49) depicts the number of records in the dataset corresponding to the year in which they were posted on Twitter. The records in the dataset comprises of data from 12 consecutive years. The bar graph depicts the maximum records present in the dataset are from the year 2020, both from the depressed and the non-depressed category, starting from the year 2008.

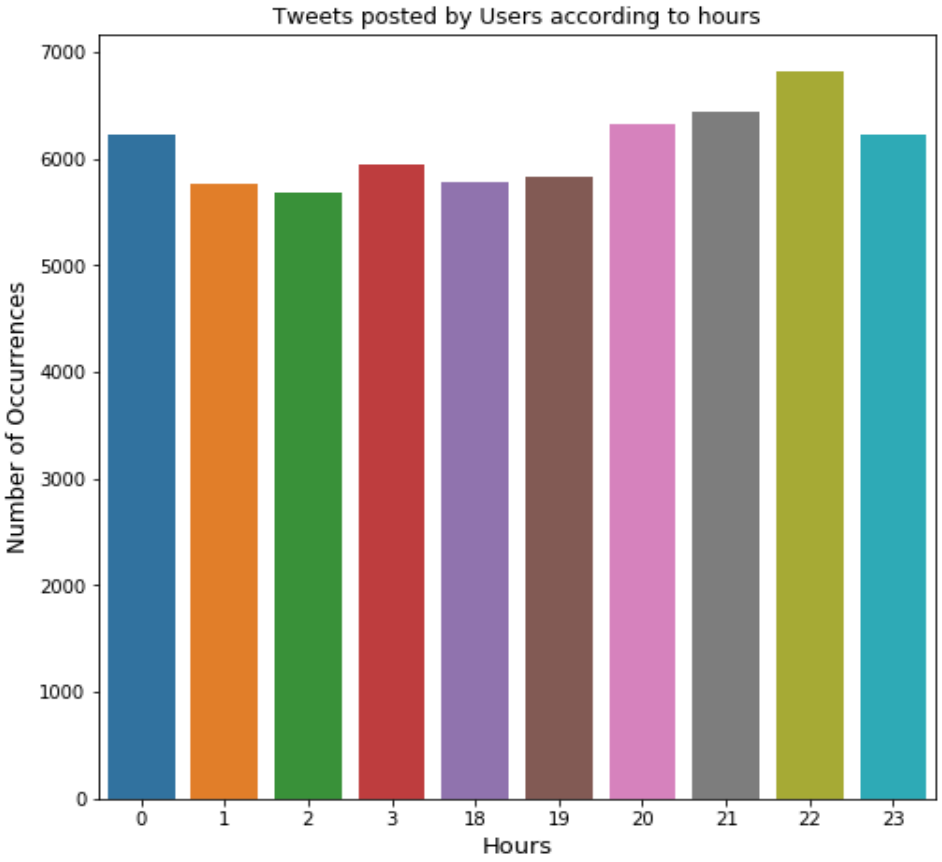


illustration 50 Showing pattern of users during busiest hours

The graph (illustration 50) states the busiest hours of the day among the both group of users, depressed and non-depressed category. The maximum tweets were posted at 10 pm from the collection of tweets from different years. The time at which maximum posts were obtained are from 6 pm to 3 am considering the most engaging time hours of users from the both the categories.

5.13 Creation of Attribute 'Age'

The age of users was calculated and added as a new feature as a part of research analysis. As per the Twitter user registration norms, the user can register himself or herself for account creation at the age of 13 [94]. Based on the hypothesis, that the person started using twitter at the age of 16, or the user has created the Twitter account at the age of 16, the value of the age attribute of each user is inferred. For instance, if the user has created the account in year 2012, then its assumed age will be (present year – the account creation year +16 [default age assumed]), which is, 2020 – 2012 +16 = 24. However, this calculation of age of users is only based upon the dataset taken under Tweet Level module of this research to study factors related with age. To add to it, the results shown in the further sections related to the comparison of age groups under this research are different than the results from stat Canada [108] because the dataset taken for this research at Tweet Level is based upon social media platform (Twitter) which differs from the dataset used by Stat Canada [108] under various factors such as source of dataset, size of dataset, location attribute of data, methodology of data collection etc.

Creation of Age Variable

```
df_vis['account_year']=pd.DatetimeIndex(df_vis['account_creation_date']).year
#creation of age variable
df_vis['age']= 2020-df_vis["account_year"]+16 #default age =16
```

```
df_vis['age'].value_counts()
```

Illustration 51 Showing a method to formulate age group

Table 57 showing users by age

Number of users by age	
Age	Users
27	15256
25	12012
24	11294
18	10486
19	9697
23	9344
26	9019
22	8702
20	8233
17	8098
21	8084
28	2678
29	586
16	222
30	34

The table 57, describes the distribution of age of users present in this dataset. The results are based upon the hypothesis discussed above, which represents the results only for the dataset used at Tweet Level for this research. The maximum number of users recorded in this dataset are of age 27, whereas the users with age 25, are second highest present in the dataset. The users with least number of age group is 30, just present 34 in number in the dataset. The average number of people in the dataset belongs to the age group 23 to 27 years.

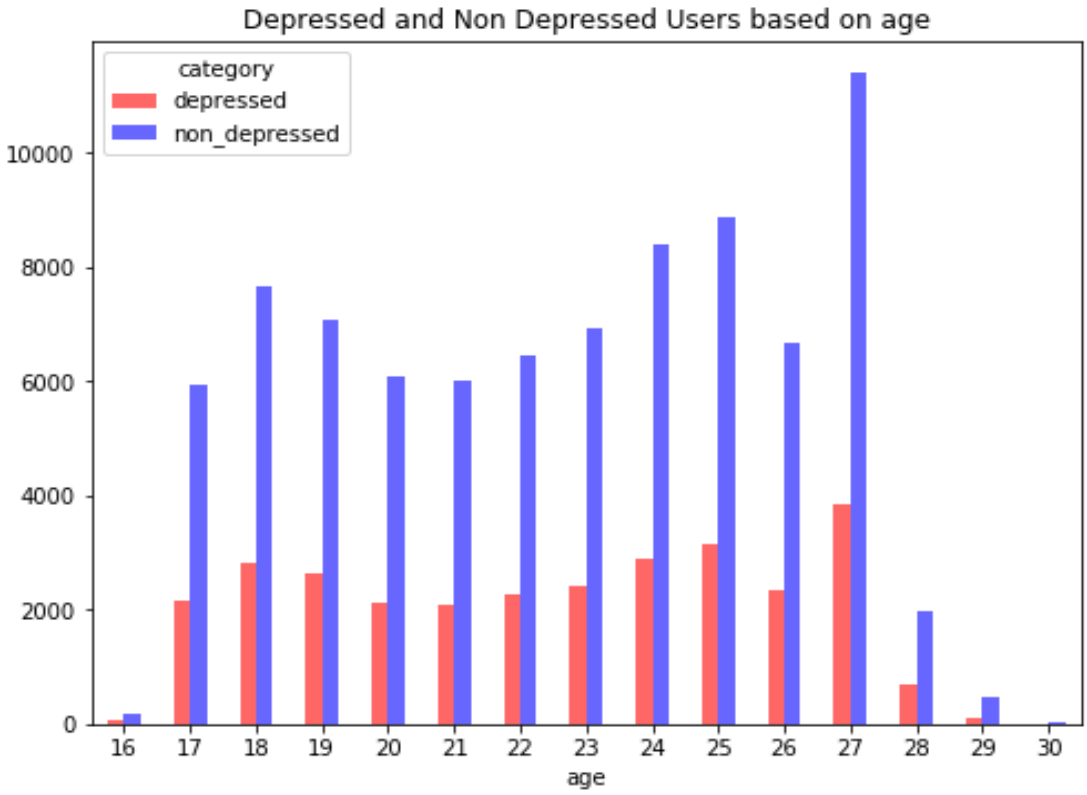


Illustration 52 Showing users category with age

The graph (illustration 52), represents the distribution of people according to their age group belonging to the depressed or non-depressed category. The analysis of relation to the people posting with their inferred age concludes that people present in the age group 24 to 27 are more prone to be affected from depression. While depression is less prevalent in the starting teenage years, and also at the age of 30, according to the age inference and user's tweets. The table 58 below also describes the exact number of users present in the dataset belonging to depressed and non-depressed category, according to the samples captured in the dataset.

Table 58 showing Twitter Posts Based on Age

Twitter Posts Based on Age		
Age	Depressed	Non-depressed
16	48	174
17	2149	5949
18	2828	7658
19	2632	7065
20	2139	6094
21	2071	6013
22	2263	6439
23	2414	6930
24	2901	8393
25	3137	8875
26	2330	6689
27	3856	11400
28	701	1977
29	121	465
30	9	25

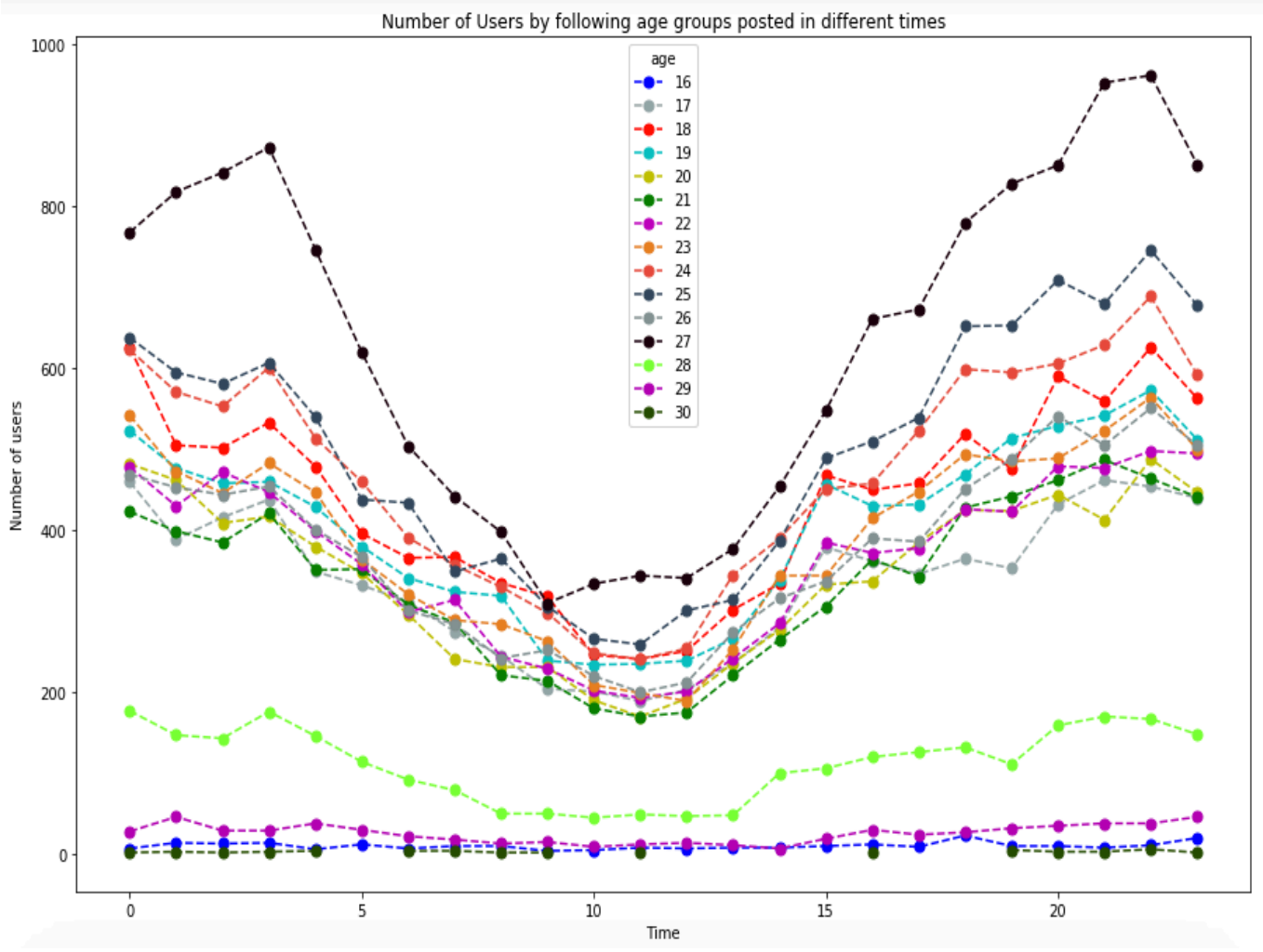
The table 58 represents the Twitter Posts based on age. The highest number of posts recorded as 11400 for Non-depressed category and 3856 for Depressed Category belong to the age group of 27, whereas the lowest number of posts are made by users in age group 30.

5.13.1 Posts Based On age of Users of depressed/Non-depressed Class

The graph (illustration 53) records the time-series analysis of the user's postings and their age groups. The graph gives insights of the number of users posted the tweets at which hour of the day and belongs to which age groups. For instance, the graph shows that around

900 users within the age group of 27 years (black line in the graph) posted their tweets between 2 am to 5 am in the morning. Similarly, people belonging to the age group 16 and 30 had least number of tweets in the dataset with regular posting pattern throughout the day.

Illustration 53 Graph Showing Correlation Between Age and Hours of Tweet Post



Chapter 6: Design and Implementation of The Proposed Solution

This chapter highlights the architecture of the proposed solution along with new methods included under the data cleaning, which targets to use the emoticons from Emojipedia [80] [81] to explore in depth emotions. Moreover, developing 12 attributes from unique methods performed using Extensive feature engineering on collected data. Also, a Correlation Matrix is being created to understand the relationship between various attributes of dataset using heatmap function. Lastly, the results are explained using various metrics for each classification.

6.1 Design

Depression is one of the most emerging mental health problems that not only impacts a person's routine but can also lead to severe conditions such as suicide. There can be scenarios, where people affected from depression, might not be themselves aware with the problem they are struggling with, but their social activities and behaviors can be brought into picture for the early detection of depression. After realizing this notion, many researchers started availing features of social media to detect the patterns of affected as well as suspected patients on social media. The context of the language or the vocabulary used by the depressed patients directly or indirectly relate to the symptoms of depression. In recent times, digitized text analysis has empowered computation of immensely huge data reserves in few minutes. According to [82], the users possess several traits while dealing with depressive symptoms in context of using social media. These traits might include, excessively keeping check on emails, using social media during peak night hours,

use of words and slangs that corresponds to the negative impact or depicting negative emotion or the virtual social circle of that person.

In today's era, people are keen to update about their lives on social media within a few minutes or seconds. Due to huge amount of utilization of social media, it has become repository of human emotions and their cognitive conceptualization. For example, if a tweet pops us saying, "Suffering from sleepless nights after my husband's death", can project to the thoughts that person is inevitably experiencing unhealthy emotional health, and could further deteriorate its mental health. According to the BMJ OPEN Study, by the researchers of the University of Pennsylvania [83], they collected and diagnosed around 400 million tweets from 2012 year to year 2016, posted by the people in their country.

They segregated the people into a bucket who have posted about 'being alone' for more than five times. Their analysis concluded that the people who reported these kinds of tweets are more inclined towards talking about their needs, relationships and family problems [83], anxiety, drinks, medications, lack of sleep, and frequently posting during night times. The study also analyzed the lexicons used by users of both the buckets. Their findings stated that people who talked more about being 'compassionate', also posted words such as 'prayers', 'amazing', 'family' whereas another group posted more about 'feel', 'myself', 'anymore' [83]. Additionally, it has been found in previous studies [1,2], that the lexical features of the language of depressed users also play a significant role in distinguishing them from people having healthy state of mind.

Depression not only disrupts the daily routine or mental peace of a person, rather its impact can be visible on the smallest of the things attempted by the person in day to day activities. Its effect can also be visible on the social media, in the form of posts the users

engage with. Not only the vocabulary, but also the sentence generation, and the grammatical structure is vital to study the effect of depression. Therefore, in this research five basic sentence level features are considered for studying the variation in depressed and non-depressed category of people. Therefore, taking inspiration and consideration of the above stated factors, the detailed architecture is described in the Illustration 54.

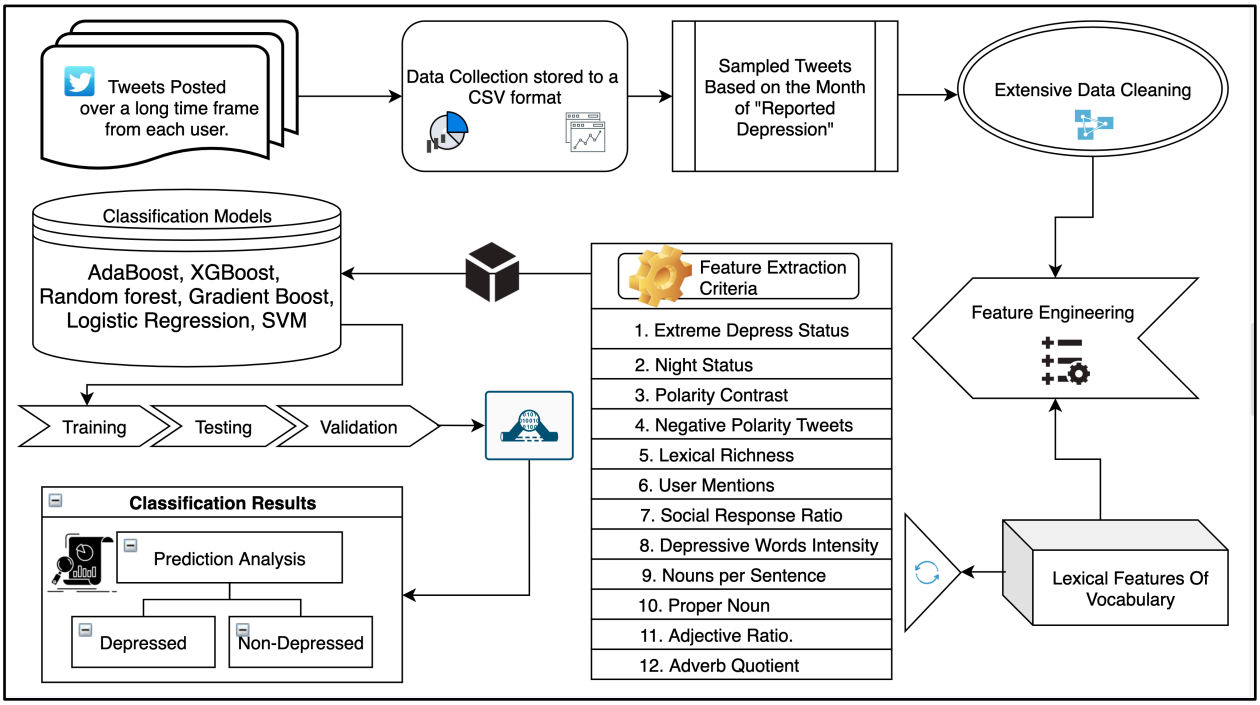


Illustration 54: Architecture of Proposed Solution

6.2 Implementation

This section represents a roadmap (Illustration 54) to be followed for the proposed solution under a user level architecture in order to reach the objective of this research within an explanation of processes working at the background. The required libraries for testing and training, as well as performing all other operations under User Level module includes Tweepy, Pandas, Emojipedia, NLTK, SciPy, BeautifulSoup, Sklearn, seaborn, Counter as

well as Textblolb. However, for the data collection, Twint- Twitter intelligence tool is required. Further, the steps involved at each stage shown in Illustration 54 are discussed as follows:

A. **Dataset Extraction:** This phase consists of the data collection of the users, who have themselves indicated the diagnosis of depression in their tweets. The dataset employed in the Experiments section, was used to obtain the list of users who belong to depressed and non-depressed class. Tweets of the users were scraped from Twitter from the recent or the latest tweet till the month in which they have specified the diagnosis of the depression. For instance, if the user has stated its diagnosis in the month of October in year 2019, then its tweets will be collected from its most recent post till the beginning of the month October 2019. Twint tool has provided the feature to scrap tweets from the user's timeline by setting the date parameter, by using the keyword '—since', that specifies the Twint time period for which it must download the data. For the formation of the dataset, relating to the tweets of the users that belong to the non-depressed category, tweets were collected from their recent post till the beginning of January 2020.

B. **Dataset Filtering for depressed Users:** From all the tweets collected for the depressed users, the tweets of each user were selected based on the month of the post for which they have posted depression diagnosis. For instance, if the user has reported depression in the month of December 2019, then all the tweets posted by the user in the month of December are filtered out to generate the profile of user. The time frame of one month is selected in order to study the detailed analysis of user behavior characteristics on social media and gives the insights of the user's activities before and after the reporting of depression.

C. **Dataset Filtering for Non-Depressed Users:** For the data or the tweets collection for the non-depressed users, all the posts of the non-depressed candidates, starting from the month of January 2020 has been filtered out. It is done to match the time frame of the depressed as well as non-depressed users, that is period of one month.

D. **Data Cleaning:** The amalgamation of large set of tweets could result to biased results if left uncleaned. Therefore, it becomes vital to clean the raw textual data to derive insightful results. The procedure of data cleaning in proposed solution is performed by following tasks:

(i) **Removal of Punctuation Marks**: The symbols that support the proper and accurate construction of sentences are referred to as Punctation marks. The functional module "String.Punctuation" available in Python is used to remove punctation marks in text. The frequent symbols of punctuation marks are " ' ( ) * ^_`{|}~ , + , - . / : ; < = > ? , ! " # $ % & , @ [\ ]" that are when encountered in the sentence, are removed from the sentence.

(ii) **Removal of HTML Tags**: Most of the real-world data extracted from web resources consists of html tags such as
, <html>, <h1>, that are inevitably adding noise to the data and increase the memory and space consumption in the phase of information Retrieval. This inoperative and futile context is removed by using 'BeautifulSoup' library that facilitates the data cleaning process by extracting data from html and xml context-based files.

(iii) **Removal of Stop Words:** Stop words can be described as the most repetitive or common words occurring depicting no significant information. These are usually ignored by search engines for retrieving search results and increases the speed of parsing web pages. List of common stop words include "and, or, not, like, could, have, want, where".

To filter out the relevant context in natural language, stop words are removed in the phase of data cleaning by traversing through each token. This could significantly decrease the size of training dataset, as well as the computation time, and promotes better classification through machine learning models.

(iv) **Removal of URL:** The links related to posts and pictures posted by the user or other retweets information that does not give insights to the sentiment of the user are removed using regular expression in NLTK.

(v) **Extracting Sentiments of emojis [80][81]:** Netizens are not limited to use just text for depicting their emotions. The research [95] indicates the usage of emoticons by users of various age groups and genders, are becoming more favorable towards usage of emoticons within the text messages or any textual information. Consequently, emoticons are also an abundant source of emotions, with each has its own specified meaning. In this research, the meaning of the emojis has been also extracted from the module Emojipedia [81], to enhance the outcome of sentiment extraction from each post of user. The table 59 adapted from Source: Unicode Emoji Charts [80][81] describes the snapshot of the emojis considered in Emojipedia [81] along with their meanings or sentiments.

Table 59 showing Emojis with their meanings[80][81]

No.	Unicode	Emoji	Sentiment
1.	U+1F970	🥰	Smiling face with hearts
2.	U+1F60D	😍	Smiling face with heart-eyes
3.	U+1F929	🤩	Star-struck
4.	U+1F618	😘	Face blowing a kiss
5.	U+1F60B	😋	Face savoring food
6.	U+1F61C	😜	Winking face with tongue
7.	U+1F61E	😞	Disappointed face
8.	U+1F620	😠	Angry face
9.	U+263A	☺️	Smiling face
10.	U+1F61A	😚	Kissing face with closed eyes
11.	U+1F613	😓	Downcast face with sweat
12.	U+1F629	😩	Weary face
13.	U+1F62B	😫	Tired face
14.	U+1F624	😤	Face with steam from nose
15.	U+1F621	😡	Pouting face
16.	U+1F603	😃	Smiling face with open mouth
17.	U+1F602	😂	Face with tears of joy
18.	U+1F609	😉	Winking face
19.	U+1F60A	😊	Smiling face with smiling eyes
20.	U+1F607	😇	Smiling face with halo
21.	U+1F622	😢	Crying face
22.	U+1F62D	😭	Loudly crying face
23.	U+1F631	😱	Face screaming in fear
24.	U+1F616	😖	Confounded face
25.	U+1F623	😣	Persevering face

E. Extensive Feature Engineering

Most of the previous researches work on predicting depression levels in users on social media by text analysis of the Tweets. However, from the gap analysis of previous research papers there is less work done on studying the pattern of user and analyzing their tweets on a time series frame. This research aims to focus on developing an extensive feature engineering criterion that will focus on language, sentence construction, lexical features of vocabulary, polarity contrast along with other features that could be analyzed from the activities of users on social media platform (Twitter). These activities include tweets, retweets, like counts, user mentions as well as reply counts for each user profile over a period of one month. Further, the activities of each user processes through the criteria under the extensive feature engineering workflow as shown in illustration 55.

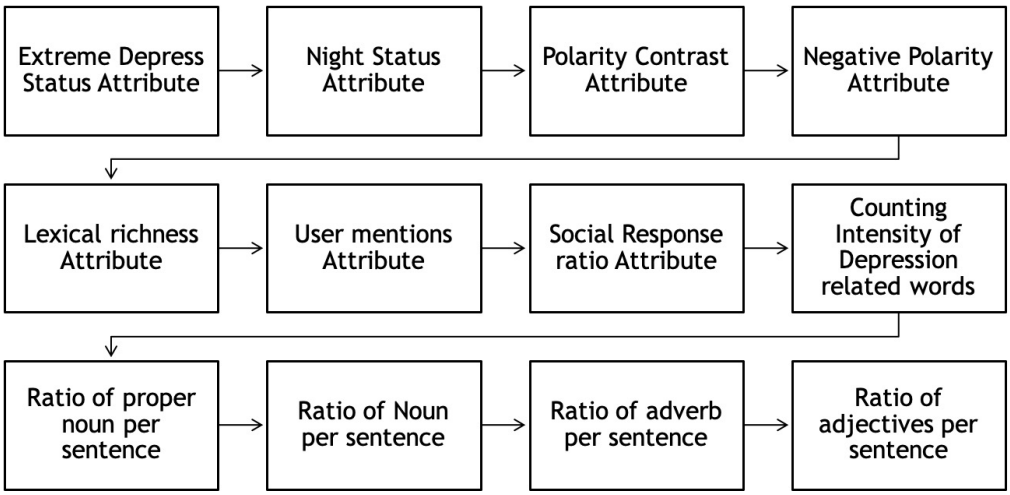


Illustration 55 showing workflow of extensive feature engineering

1. **Extreme Depress Status Attribute:** The social media user suffering from depression, tend to use words that are closely related to depression, anxiety or other similar words that

depict their unhealthy mental state. According to [89], the words similar to depression and also incorporated in this research to determine this attribute are:

Table 60 showing depression related words [89]

'illusion', 'restless', 'bored', 'crap', 'shit', 'sad', 'escape', 'useless', 'meaningless', 'suffer', 'suffering' , 'sleepless', 'never', 'bored', 'cry', 'afraid', 'unhappy', 'ugly', 'upset', 'awful', 'torture', 'unsuccessful', 'helpless', 'suffer', 'fail', 'sorrow', 'nobody', 'blame' , 'damaged', 'shatter' , 'Fat', 'bad', 'weak', 'problem', 'tired' ,'pathetic', 'insomnia', 'kill', 'panic', 'lonely', 'hate', 'depressed', 'frustrated', 'loser', 'suicidal', 'hurt', 'reject', 'painful', 'disappoint', 'broke', abandon', 'worthless', 'regret', 'dissatisfied', 'lost', 'empty', 'destroy', 'ruin', 'die', 'sick', 'depression' , 'anxious', 'trauma'.

In this research, if the user has posted more than five tweets containing any of the word from the above-mentioned list [89], then the attribute value corresponding to that particular user is set '1' (indicating symptoms of depression) otherwise '0' (non-depressive symptoms).

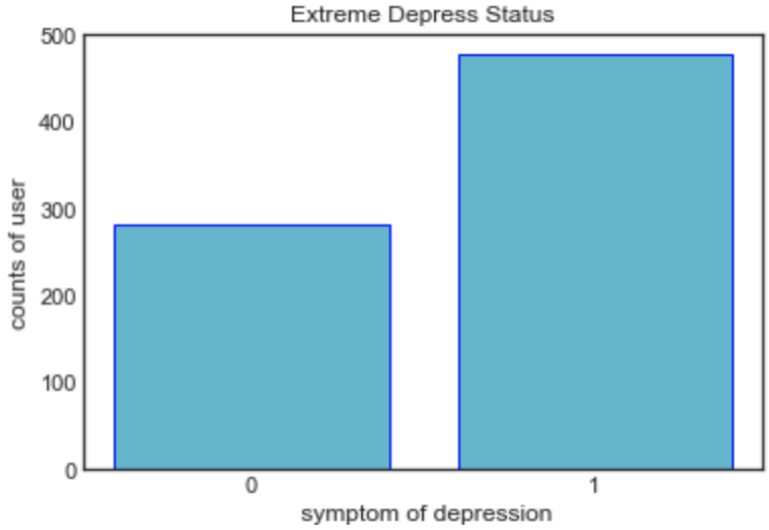


Illustration 56 showing outcomes for extreme depress status attribute

The illustration 56 shows the outcomes of 760 users for Extreme Depress Status Attribute. The number of users having value 1 shows that they have posted Tweets indicating symptoms of depression, whereas, the value for remaining users who have not posted any tweets related to symptoms indicating depression was set to 0.

2. **Night Status Attribute:** To understand the notion of emotions, present in the text, current neurosciences theory classifies emotions into two types: Categorical and Dimensional [86]. The category-based approach introduced by Darwin states that these kinds of emotions consists of countable kind of emotions that are acknowledged worldwide [86]. The second kind of emotions distribute each emotional state into multiple dimensions, and then define whether the emotion belongs to the positive or negative axis [86]. There has been quite extensive research work conducted in order to find the correlation between the sleep hours and the frequency of time engagement by a user on social media in night hours [85]. It has been found that the sleep patterns of users are prodigiously monopolized by social media usage. To add to it, the properties exhibited by the content, whether it is positive or negative equally plays an important role in determining the effect of depression on the sleeping pattern of the user [85]. In this research, the polarity of the posts has been taken into consideration. If the user has user has posted more than two tweets in midnight hours (11 pm to 6 am) that stipulates negative emotion of the tweet, then the value of night status for that particular user is set to be '1'. The polarity of the tweets is detected by using 'TextBlob' [90], python library inbuilt for sentiment analysis. It predicts the output in the scale of -1 to +1 to determine the polarity of context, with -1 (words such as suicide, kill) being extreme negative, +1 being extreme positive (for instance, amazing, cheerful) , and

0 being neutral (example: should). The flowchart (Illustration 57) below describes the phenomena by which the value of attribute night status is calculated.

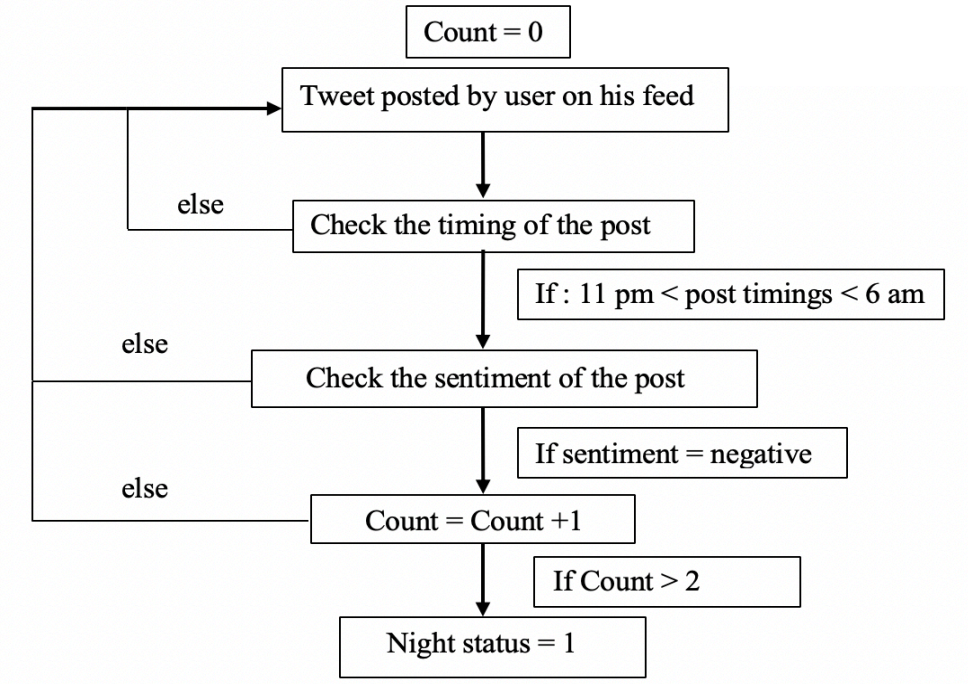


Illustration 57 showing creation of attribute night status

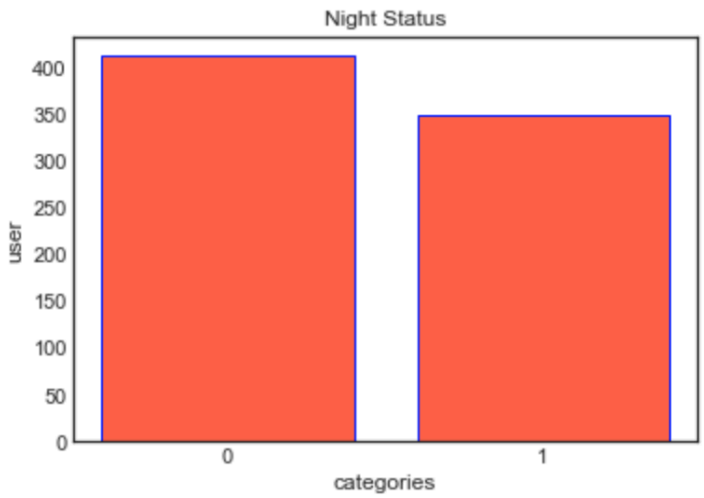


Illustration 58 showing outcomes for night status attribute

The illustration 58 represents the outcomes of Night Status Attribute for 760 users. The users with value 1 show that they have posted more than 2 negative tweets during midnight

hours (11pm to 6am). However, the value for the remaining users who did not follow the workflow as shown in illustration 57 was set as 0 .

3. **Polarity Contrast Attribute:** There have been times, when user posts at 1 p.m. "Enjoying lunch at Japanese restaurant", but at night the same user posts, "Not able to sleep, feeling alone, hoping this life to end". Therefore, the shift of the polarity in between the sentences can be a good indicator to diagnose the depression. Moreover, such regular switches in polarity points to the fluctuation and discrepancies in healthy state of mind and can be considered as the symptom of depression [89]. The attribute overall sentiment polarity indicates the overall alteration of the polarity of the posts and can be calculated by an equation adapted from [89] as shown:

$$\textbf{(Polarity Contrast) PC} = \frac{(ß * Post\_pc + pos\_wc) - (ß * Post\_nc + neg\_wc)}{(ß * Post\_pc + pos\_wc) + (ß * Post\_nc + neg\_wc)}$$

Where, Post_pc = count of positive posts

Post_nc = count of negative posts

Pos_wc = count of positive words

neg_wc = count of negative words

ß = hyper-parameter, depending on number of posts, in this case = 4

Furthermore, there are some situations under which a user posts something negative but does not really mean it, which could result in false outcome. Such types of situations can be controlled using the Polarity Contrast Attribute which calculates the shift of polarity by

balancing out the positive and negative outcomes from all tweets of each user by using the above equation for Polarity Contrast. This equation will help to distinguish between the change of the positive sentiments as well as the negative sentiments used under the Tweets by each user.

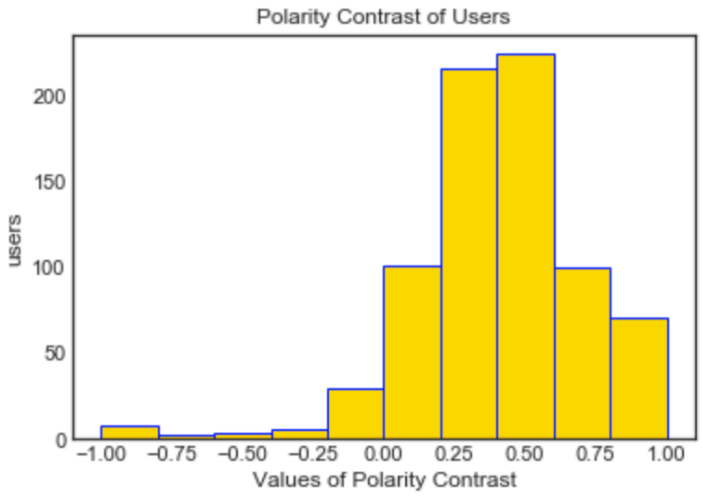


Illustration 59 showing outcomes for polarity contrast attribute

The illustration 59 represents the outcomes of Polarity Contrast Attribute for 760 users. The values ranging from -1 to + 1 represents the shift of polarity for each user. However, only a few users can be observed with negative shift of polarity and the remaining users were holding positive values.

4. **Negative Polarity Attribute:** The polarity of the context plays a vital role when dealing with mental health problems. According to [91], depressed people tend to use negative words or write sentences that indicate negative emotion more than the average user. Therefore, it is necessary to consider overall polarity of the sentences or tweets posted by the user to diagnose how often the user tweets, conveying negative meaning. In this research, the attribute negative polarity is a categorical variable, that is set to '1', if user has more than 20% posts that expressed negative emotions, otherwise it is set to '0'. The

sentiment of the sentence is calculated by using TextBlob [90], which indicates responses in three emotions, positive, negative, and neutral.

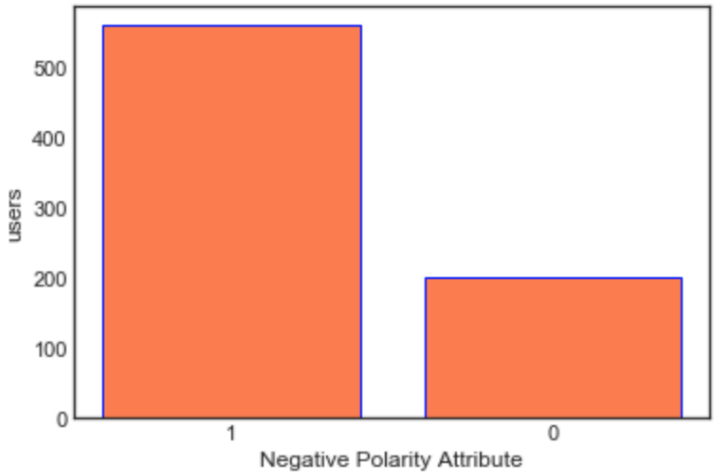


Illustration 60 showing outcomes for negative polarity attribute

The illustration 60 shows the outcomes of Negative Polarity Attribute for 760 users. Out of all posts made by a user, if more than 20% posts express negative emotions, then the value for a user is set to 1, otherwise 0.

5. **Lexical richness Attribute:** The attribute lexical richness demonstrates the amount of pertinent information present in the context. In other words, it can be defined as the ratio of unique words in the sentence to the total words in the sentence. In this research, the lexical richness is tweaked in such a way, that the lexical richness is not just limited to the sentence level or tweet level rather it has considered all the past tweets of the user from one month and prepared the vocabulary of each user. The lexical richness, corresponding to the user is then calculated by dividing the total number of unique words in the user vocabulary to the total number of words in the tweets posted by the user within one month. It describes how capable is the user to share his/her thoughts in more coherent manner.

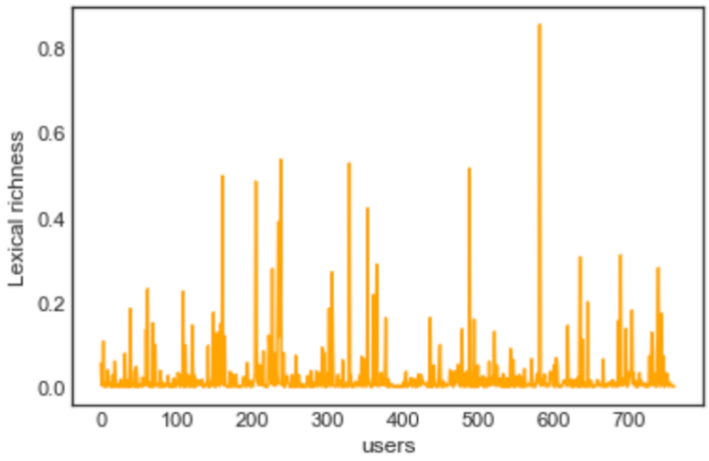


Illustration 61 showing outcomes for lexical richness

The illustration 61 shows the outcomes of Lexical Richness Attribute for 760 users. The y axis shows the ratio of lexical richness recorded for each user and the x axis shows the number of users. It helps to understand that most of the people score ratio under 0.1 which means that they have used less unique words as compared to users with higher ratio.

6. **User mentions Attribute:** It is a prevalent practice, usually when user posts something on social media interfaces, he or she wants to express himself/herself to other users directly. Twitter provides the feature to the user to include the names of other fellows added on social media circle to mention or tag their names along with the post. User mentions in twitter terminology indicates the names of other users or social media peers that a person includes, while posting a tweet on twitter. 'Twint' or twitter intelligence tool provides the feature of displaying user mentions per tweet the user has posted. The attribute user mentions in this research indicates the numeric value stating the average user mentions per tweet of the user, in that following month. In other notation, the attribute user mentions in this research can be defined as:

User mentions = Sum of user mentions in all tweets of a user

$$\textbf{User mentions ratio (per tweet)} = \frac{\text{Sum of number of user mentions in all tweets of a user}}{\text{Total number of tweets of user in one month}}$$

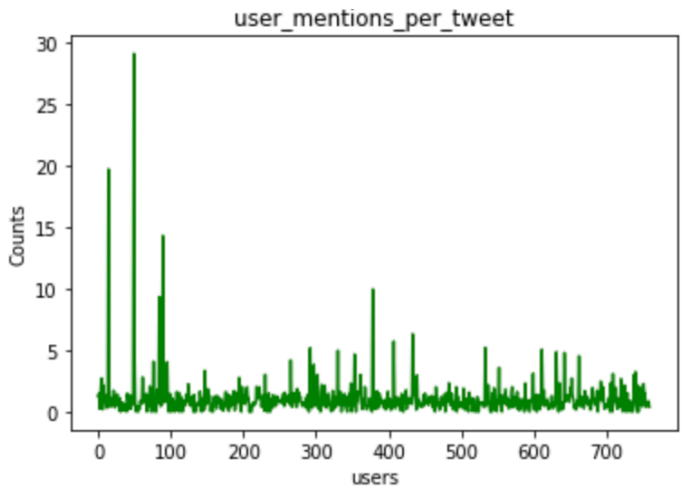


Illustration 62 showing outcomes for user mention attribute

The illustration 62 Shows the outcomes for User Mention Attribute of 760 users. The y axis shows the user mention ratio per tweet for each user and the x axis shows the number of users. However, it can be observed that most of the users tend to mention 2-3 users per tweet and the remaining users with value 0 shows that they did not mention any user in their tweets. Moreover, only a few users were recorded, that mentioned more than 4 users in their Tweets.

7. **Social Response ratio Attribute:** Social media activities of a user can be investigated by keeping a check on various parameter or by auditing his/her usage of particular features. To analyze the user engagement at certain level on social media, the features encoded in between the social networking sites such as likes, comments, replies on certain posts, sharing rate, can be proactively exploited in order to discover the general responsiveness of the user on virtual or social network. In this research, the attribute social response ratio takes into consideration the features, number of replies the user has received on its post, how many other users have shared the exact same post, and the number of likes the user

has received on its post. These features are already compiled with the twitter user profile and can be easily obtained by Twint. The formula used to calculate social response ratio in this research is:

Social Response ratio = (replies count + retweets count + likes count) for all the posts over a month

$$\text{Social Response ratio per tweet} = \frac{\text{(replies count + retweets count + likes count) for all the posts over month}}{\text{Total number of tweets of user in one month}}$$

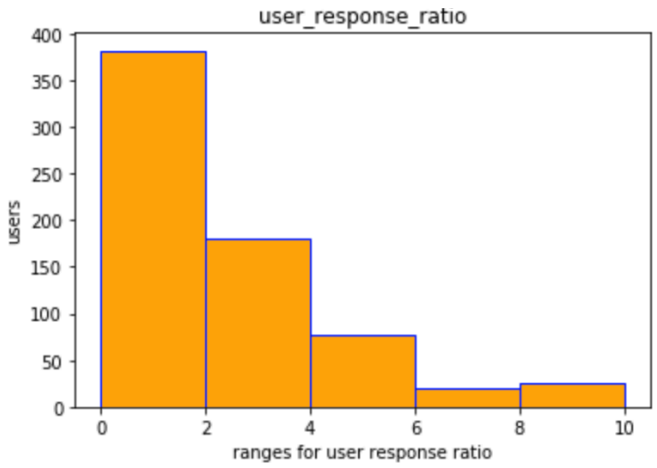


Illustration 63 showing outcomes for Social response ratio attribute

The illustration 63 shows a graph using buckets (or bins) to represent the range of outcomes of 760 users for Social Response Ratio Attribute. The y axis shows the number of users and the x axis shows the range of social response ratio per tweet. However, it is easier to understand a graph using bins rather than plotting a single value for each user which makes it very complicated to observe outcomes from visualization. Further, most of the people recorded a social response ratio per tweet under range of 0 to 2 and only a few users tend to have a social response ratio per tweet under range 8 to 10 followed by users under range 6 to 8.

8. **Counting Intensity of Depression related words:** For depressed users, the negative thoughts are not just limited to 1 or 2 posts but extended to multiple posts. Therefore, it is

necessary to keep check how many times, user has talked about words similar to depression. To calculate the value of this attribute, all the tweets of the users are traversed and the number of occurrences of depression related words are counted by comparing the terms in all tweets with the above stated depression keywords list (table 60), and then the sum of all occurrences of depression related words is divided by the total number of tweets posted by the user in one month. For instance, if a user has posted 50 tweets in a month, and contains depression related keywords in 25 tweets (considering single depression related term per tweet), then the value of this attribute will be 25/50 = 0.5, which indicates that on an average, the user has posted depression context of 50% in his tweets.

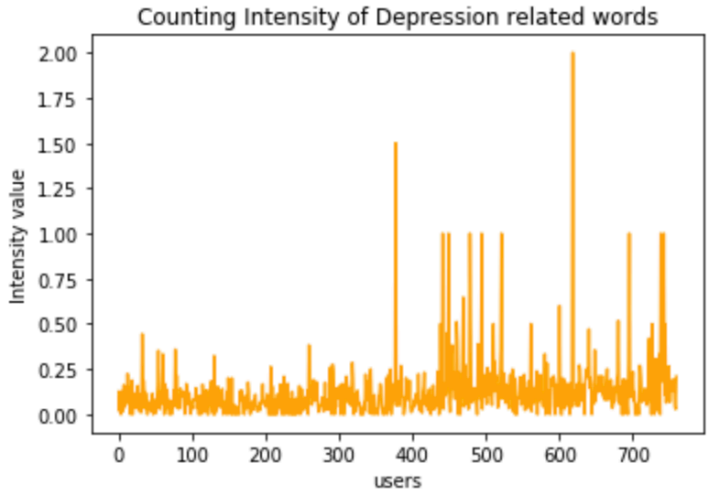


Illustration 64 showing outcomes for counting intensity of depression related words

The illustration 64 shows outcomes for Counting Intensity of Depression Related Words. The x axis shows the number of users and the y axis represents the value of intensity of depression related words. However, if a user has used depression related keywords more than the number of Tweets the Intensity value will be above 1. However, the users with intensity value 0 shows that they have not used any depression related keywords in any of their tweets over a period of one month.

9. **Ratio of proper noun per sentence:** The attribute audits the number of nouns used by the user while expressing his thoughts in the form of tweets. The values of this attribute contain the ratio of the total number of existences of proper noun in each user's vocabulary, which is built to the total number of tweets posted by the user in one month. This attribute aids to understand the variation in sentence structure and how opinionatedly the users of both categories (depressed classes and non-depressed classes) can employ the grammatical structures and follow the grammatical rules while writing the sentences. The NLTK module for POS tagging is used for identifying the proper nouns in sentences. It assigns the label 'NNP' to the token that belongs to the category of proper noun.

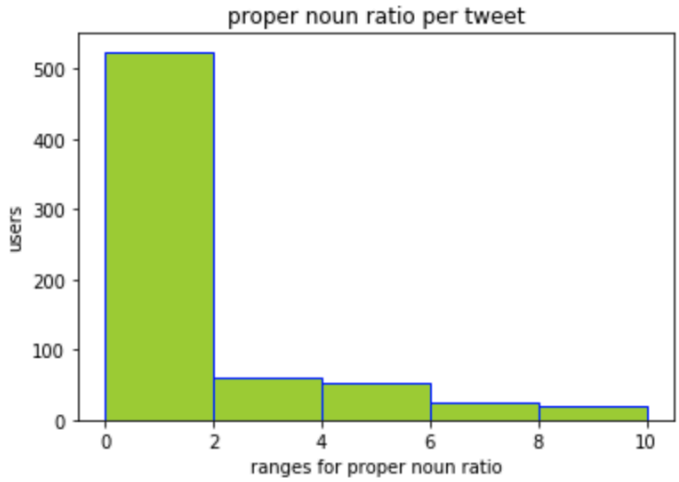


Illustration 65 showing ratio of proper noun per sentence

The illustration 65 shows the visualization for Ratio of Proper Noun Per Sentence. The y axis shows the number of users and the x axis shows the range for ratio of proper noun per tweet for each user. Further, most of the users tend to use proper noun per tweet under a range of 0 to 2. However, only a few users tend to use more than 2 proper pronouns in their tweet over a period of one month.

10. **Ratio of Noun per sentence:** The attribute calculates the ratio of nouns used by the user per tweet. Nouns are the vital part of sentence creation, referring to different objects

or entities. This attribute will enable to give insights, how frequently depressed and non-depressed users refers to other entities while expressing their emotions on twitter. The NLTK's POS tagging is used to deliberately assigning the labels to the tokens. The token 'NN' indicated the noun in the sentence.

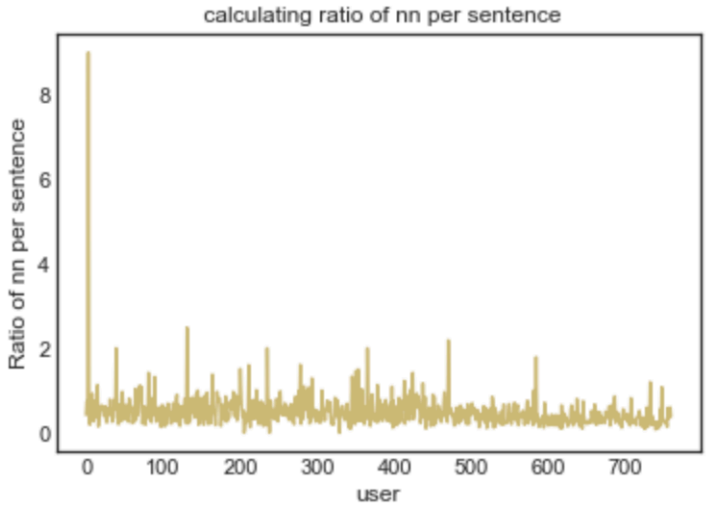


Illustration 66 showing ratio of noun per sentence

The illustration 66 shows the visualization for Ratio of Noun Per Sentence. The x axis shows the number of users and the y axis shows the ratio of noun per sentence. Further, it can be observed that most of the people have used 1 noun per sentence and the remaining users with value 0 have used no nouns at all. Moreover, only a few users have used more than 2 nouns in their tweet over a period of one month

11. **Ratio of adverb per sentence:** This attribute contains the numeric value indicating the ratio of adverbs used by the user over the timeline of tweets of one month. Depressed users are more inclined towards expressing their feelings with usage of adverbs by putting more weight on the situation or by exaggerating the quality of the context. Therefore, this ratio is calculated by considering all the adverbs in each user's

vocabulary. The POS tagging is used to segregate the adverb from other words of the sentences. It assigns tag ‘RB’ to the adverb.

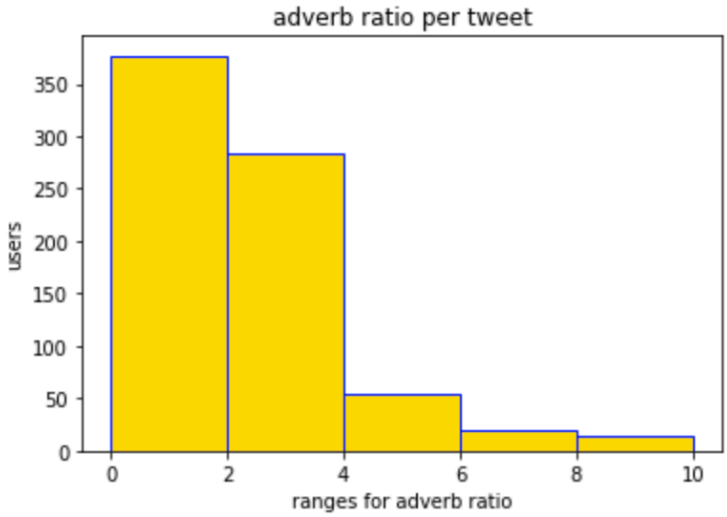


Illustration 67 showing outcomes for ratio of adverb per sentence

The illustration 67 shows the outcomes for Ratio of Adverb Per Sentence. The y axis represents the number of users and the x axis shows the range of adverb ratio per sentence. Further, it can be observed that most of the users tend to use adverbs in their tweets under a range of 0 to 2 followed by users under a range of 2 to 4. However, only a few users have used more than 4 adjectives in their tweet over a period of one month.

12. **Ratio of adjectives per sentence:** This attribute contains the ratio of the adjectives used by the user while expressing their emotions over twitter. This attribute aids to know the deep insights of the variance of grammatical structures in the sentences of depressed and non-depressed users in the form of adjectives, as this part of speech can also denote the quality of the person or thing as well as for comparison. People prone to depression are habitual to compare themselves or possess the inferiority complex, to which they adapt adjectives to communicate their ideas. The POS tag ‘JJ’ denotes the adjective in the sentence when passed through the procedure of POS tagging.

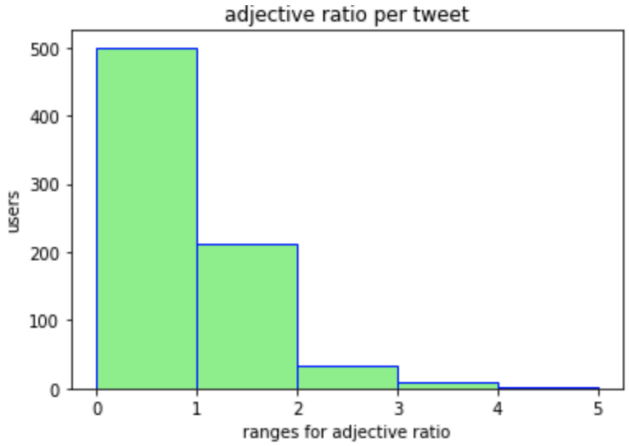


Illustration 68 showing outcomes for ratio of adjective per sentence

The illustration 68 shows the outcomes for Ratio of Adjectives Per Sentence. The x axis shows the range of adjective ratio per sentence and the y axis shows the number of users. Further, it can be observed that most of the users tend to use adjectives in their tweets under a range of 0 to 1 followed by 1 to 2. However, only a few users have used more than 2 adjectives in their tweet over a period of one month.

Nevertheless, after the feature engineering, all the outcomes are integrated into a single data frame. This data frame will be further used by machine learning algorithms for training and testing. The pattern learned by the classification models from the outcomes of feature engineering criteria will ultimately predict the desired output.

6.2.1 System Model for Proposed Solution

The system model for proposed solution will explain the steps involved in transforming tweets of a user into a useful set of information. This information will be processed under feature engineering criteria to analyze pattern of user engagement and predict depression in users on social media (Twitter). The tweets from users will be converted into a final

matrix (data frame) using various operations discussed further in this section. The data frame will be processed further as input for the classification models.

I. Let U be the set of m number of users (U) = (U_1,U_2, U_3…… U_m)

II. Each user will have different number of tweets posted for a period of one month. Therefore, let (TS) be equal to the set of n number of tweets (t) posted by a user in one month, where the value for n will change according to the user.

III. Now for user U_1, the number of Tweets(t) will be (TS) = {t_1, t_2, t_3…..tn}.

IV. Let W be the set of z number of words (w) in (TS), W = {w_1, w_2, w_3……w_z}

V. Let LT be the set of depression related words, (LT) = {'illusion', 'restless', 'bored', 'crap', 'shit', 'sad', 'escape', 'useless', 'meaningless', 'suffer', 'suffering' , 'sleepless', 'never', 'bored', 'cry', 'afraid', 'unhappy', 'ugly', 'upset', 'awful', 'torture', 'unsuccessful', 'helpless', 'suffer', 'fail', 'sorrow', 'nobody', 'blame' , 'damaged', 'shatter' , 'Fat', 'bad', 'weak', 'problem', 'tired' ,'pathetic', 'insomnia', 'kill', 'panic', 'lonely', 'hate', 'depressed', 'frustrated', 'loser', 'suicidal', 'hurt', 'reject', 'painful', 'disappoint', 'broke', abandon', 'worthless', 'regret', 'dissatisfied', 'lost', 'empty', 'destroy', 'ruin', 'die', 'sick', 'depression' , 'anxious', 'trauma'}

The steps involved for feature engineering criteria are as following:

1. **Extreme Depress Status Attribute:** For a user U_1, if **TS = {t_1, t_2, t_3…..tn} > 5 tweets** containing any of the word from LT, then the attribute value corresponding to that particular user (U_1) is set '1' (indicating symptoms of depression) otherwise '0' (non-depressive symptoms).

2. **Night Status Attribute:** For a user U_1, if **TS = {t_1, t_2, t_3…..tn} > 2 tweets** in midnight hours (11 pm to 6 am) that stipulates negative emotion of the tweet, then the value of night status for user (U_1) is set to be '1', otherwise '0'.

3. **Polarity Contrast Attribute:** The attribute overall sentiment polarity indicates the overall alteration of the polarity of the posts and can be calculated by an equation adapted from [89] as shown:

For a user U_1, the set of n tweets posted in one month period **TS = {t_1, t_2, t_3…..tn}**

Let W be the set of z words for (TS), W= {w_1, w_2, w_3……w_z}

Let the number of negative tweets in (TS) = post_nc

Let the number of positive tweets in (TS) =post_pc

Let the number of positive words in (W) = post_wc

Let the number of negative words in (W)= neg_wc

Let ß be the hyper-parameter depending on number of posts, for this research the value was set as (ß)= 4

$$\textbf{(Polarity Contrast) PC} = \frac{(ß * Post\_pc + pos\_wc) - (ß * Post\_nc + neg\_wc)}{(ß * Post\_pc + pos\_wc) + (ß * Post\_nc + neg\_wc)}$$

4. **Negative Polarity Attribute:** For a user U_1, if **TS = {t_1, t_2, t_3…..tn} > 20% posts that express negative emotions**, then the value for negative polarity of the user (U_1) is set to 1, otherwise 0. However, the sentiment of the sentence is calculated by using TextBlob [90], which indicates responses in three emotions, positive, negative, and neutral.

5. **Lexical richness Attribute:** The lexical richness, corresponding to the user is calculated by dividing the total number of unique words in the user vocabulary to the total number of words in the tweets posted by the user within one month. It describes how capable is the user to share his/her thoughts in more coherent manner.

For a user U_1, where TS = {t_1, t_2, t_3…..tn} and W={w_1, w_2, w_3……w_z}

Let UW be the number of y unique words in W, therefore UW = { uw_1,uw_2, ….uw_y}

$$\textbf{Lexical richness Attribute} = \frac{\textbf{Total number of words in UW}}{\textbf{Total number of words in W}}$$

6. **User mentions Attribute:** The attribute user mentions in this research indicates the numeric value stating the average user mentions per tweet of the user, in one month of tweets posted. In other notation, the attribute user mentions in this research can be defined as follows:

Let UM be the set of k number of user mentions (um) in total tweets (TS) for a user (U_1). Therefore, UM = $\{um_1, um_2, \ldots um_k\}$

Total User mentions = Sum of number of user mentions in (UM)

$$\textbf{User mentions ratio} = \frac{\textbf{Total User mentions}}{\textbf{Total number of tweets in (TS)}}$$

7. **Social Response ratio Attribute:** The formula used to calculate social response ratio in this research is as follows:

Let the total number of replies the user received on (TS) tweets posted in one month=RPC

Let the total number of retweets the user received on (TS) tweets posted in one month=RTC

Let the total number of likes the user received on (TS) tweets posted in one month=LC

Total Social Response = (replies count + retweets count + likes count) for all the posts over a month, or **Total Social Response = RPC + RTC + LC**

$$\textbf{Social Response Ratio for a user} = \frac{\textbf{Total Social Response}}{\textbf{Total number of tweets in (TS)}}$$

8. **Counting Intensity of Depression related words:** To calculate the value of this attribute, all the tweets of the users are traversed and the number of occurrences of depression related words are counted by comparing the terms in all tweets with the above

stated depression keywords list (LT), and then the sum of all occurrences of depression related words is divided by the total number of tweets posted by the user in one month. For instance, if a user has posted 60 tweets in a month, and contains depression related keywords in 30 tweets (considering single depression related term per tweet), then the value of this attribute will be 30/60 = 0.5, which indicates that on an average, the user has posted depression context of 50% in his tweets.

Let DW be the set of occurrences of words calculated by matching the words between (W) and the words in list (LT). Therefore,

Counting Intensity of Depression related words = Sum of values in DW/ Total number of tweets in (TS)

9. **Ratio of proper noun per sentence:** The attribute audits the number of nouns used by the user while expressing his thoughts in the form of tweets. The values of this attribute contain the ratio of the total number of existences of proper noun in each user's vocabulary, which is built to the total number of tweets posted by the user in one month. This attribute aids to understand the variation in sentence structure and how opinionatedly the user can employ the grammatical structures and follow the grammatical rules while writing the sentences. The NLTK module for POS tagging is used for identifying the proper nouns in sentences.

Let PN be the set of d number of proper pronouns (Pn) in (W), therefore PN = $\{Pn_1, Pn_2, \ldots Pn_d\}$

Total Proper Nouns = Sum of number of values in PN

Ratio of proper noun per sentence = Total number of tweets in (TS)/ Total Proper Noun

10. **Ratio of Noun per sentence:** The attribute calculates the ratio of nouns used by the user per tweet. Nouns are the vital part of sentence creation, referring to different objects or entities. This attribute will enable to give insights, how frequently depressed and non-depressed users refers to other entities while expressing their emotions on twitter.

Let NN be the set of e number of nouns (Nn) in (W), therefore NN = $\{Nn_1, Nn_2, \ldots Nn_e\}$

Total Nouns = Sum of number of values in NN

Ratio of noun per sentence = Total number of tweets in (TS)/ Total Nouns

11. **Ratio of adverb per sentence:** This attribute contains the numeric value indicating the ratio of adverbs used by the user over the timeline of tweets of one month. Some users are more inclined towards expressing their feelings with usage of adverbs by putting more weight on the situation or by exaggerating the quality of the context. Therefore, this ratio is calculated by considering all the adverbs in each user's vocabulary.

Let RB be the set of h number of adverbs (Rb) in (W), therefore RB = $\{ Rb_1, Rb_2, \ldots Rb_h\}$

Total Adverbs = Sum of number of values in RB

Ratio of Adverb per sentence = Total number of tweets in (TS)/ Total Adverbs

12. **Ratio of adjectives per sentence:** This attribute contains the ratio of the adjectives used by the user while expressing their emotions over twitter. This attribute aids to know the deep insights of the variance of grammatical structures in the sentences of depressed and non-depressed users in the form of adjectives, as this part of speech can also denote the quality of the person or thing as well as for comparison. People prone to depression are habitual to compare themselves or possess the inferiority complex, to which they adapt adjectives to communicate their ideas.

Let AJ be the set of i number of adverbs (Aj) in (W), therefore AJ = $\{ Aj_1, Aj_2, \ldots Aj_i\}$

Total Adjectives = Sum of number of values in AJ

Ratio of Adjective per sentence = Total number of tweets in (TS)/ Total Adjectives

Nevertheless, after the feature engineering, all the outcomes are integrated into a single data frame. This data frame will be further used as an input to the machine learning algorithms for training and testing. The pattern learned by the classification models from the outcomes of feature engineering criteria will ultimately predict the outcome for desired target variable as discussed under section 6.3.

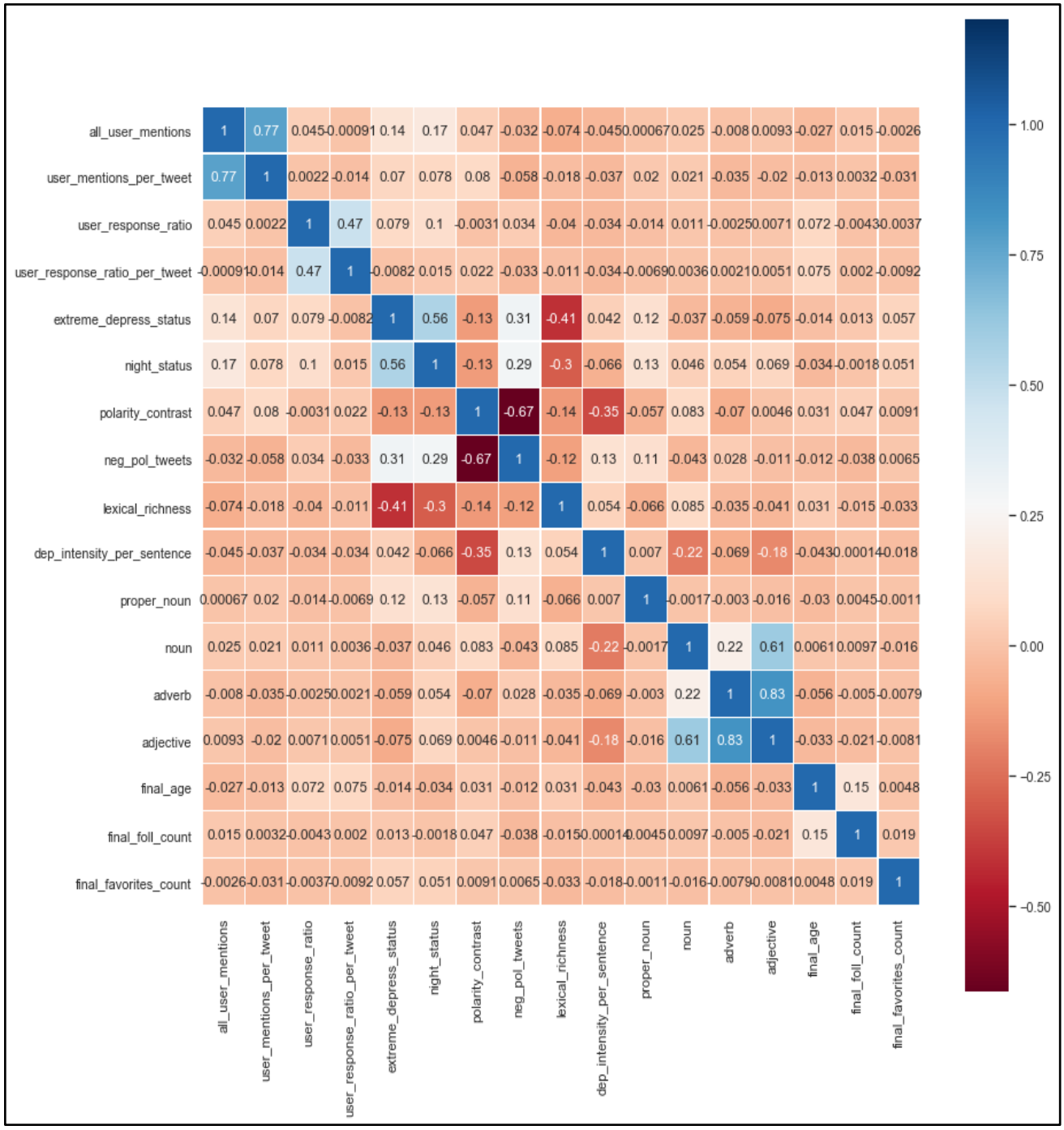
The Illustration 69 represents the sample of final output (data frame) generated by extensive feature engineering criteria that will be used by classification models for testing and training. However, the rows represent the features whereas the columns represent the outputs of users.

	0	1	2	3	4	5
all_user_mentions	20.000000	403.000000	24.000000	11.000000	829.000000	442.000000
user_mentions_per_tweet	1.250000	1.476190	0.240000	1.222222	2.355114	2.711656
user_response_ratio	21.000000	4316.000000	171.000000	4.000000	3503.000000	452.000000
user_response_ratio_per_tweet	1.312500	15.809524	1.710000	0.444444	9.951705	2.773006
extreme_depress_status	0.000000	0.000000	1.000000	0.000000	1.000000	1.000000
night_status	0.000000	0.000000	1.000000	0.000000	1.000000	0.000000
polarity_contrast	0.016949	0.738636	0.176000	0.714286	0.348035	0.698482
neg_pol_tweets	1.000000	0.000000	1.000000	0.000000	1.000000	0.000000
lexical_richness	0.056140	0.004927	0.013676	0.108333	0.001868	0.002139
dep_intensity_per_sentence	0.125000	0.018315	0.100000	0.000000	0.127841	0.067485
proper_noun	0.640000	0.350000	0.520000	0.210000	0.380000	0.250000
noun	0.420000	0.730000	0.550000	9.000000	0.270000	0.190000
adverb	1.140000	5.570000	2.860000	9.000000	1.740000	0.540000
adjective	0.700000	2.530000	1.250000	9.000000	0.550000	0.330000
final_age	17.000000	22.000000	21.000000	23.000000	21.000000	21.000000
final_foll_count	38.000000	840.000000	272.000000	144.000000	124.000000	373.000000
final_favorites_count	1.984000	46.040000	18.829000	0.145000	12.523000	3.750000

Illustration 69 Sample of output for final data frame

Correlation matrix: A correlation matrix helps to find the association or inter connection between different attributes. Also, higher the value of correlation coefficient, more is the bonding between attributes.

Illustration 70 showing correlation between various attributes



6.3 Results of all classification models for User to User Level

After the feature engineering, the outputs are then operated into a data frame. This data frame is used as input for the classification models on a 50% training and a 50% testing split of the data. The results from each classification models are further validated by K-fold scores for each model under section 7.4. Nevertheless, this section represents the results of all the models used under the proposed solution on user level architecture by applying mathematical operations on confusion matrix to reveal various outputs as per the metrics defined, followed by plotting of the ROC curve (area under the curve).

6.3.1 Evaluation of Confusion Matrix for Random Forest for User Level

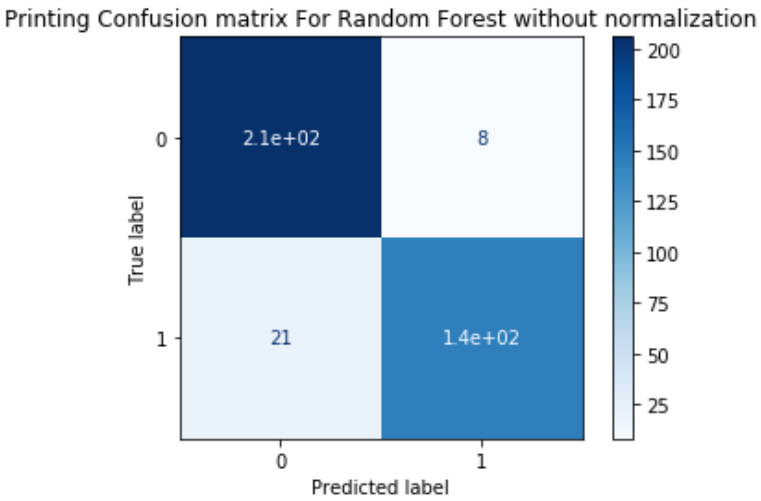


Illustration 71 Showing Confusion Matrix without normalization for random forest for user level

The illustration 71 shows a confusion matrix without normalization for Random Forest at User Level. Each value in the matrix shows the total number of instances in the testing data in the form of an equation. In order to view the exact value of each predicted instance the equation gets converted into numbers printed by the classifier as shown in Table 61

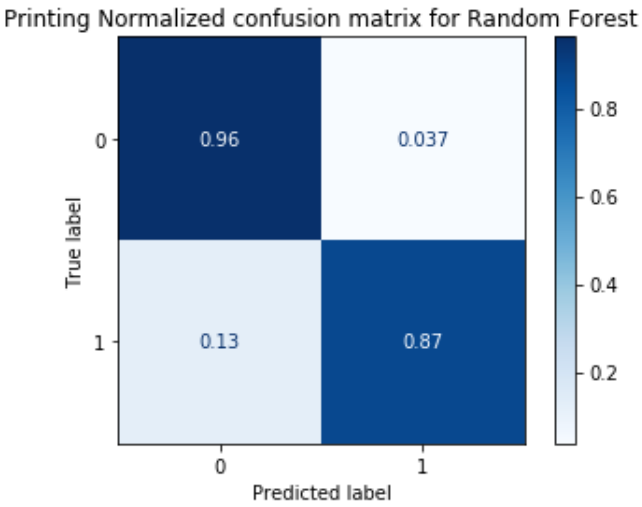


Illustration 72 Showing Normalized Confusion Matrix for random forest for user level

The Illustration 72 shows a normalized confusion matrix for Random Forest at User Level. The decimal representation helps to understand the results by observation of the diagonal values showing the degree of accurately predicted occurrences, which are higher than the off diagonal values. Moreover, the sum of values of each row is equal to 1.00 that shows the 100% utility of possible outcomes for each class.

Table 61 showing confusion matrix for Random Forest at user level

	Depressed	Non-Depressed
Predicted as Depressed	TP = 145	FP = 8
Predicted as Non-Depressed	FN = 21	TN = 206

The Table 61 Shows the values of all possible predicted outcomes for Random Forest at User Level. The values in the matrix are explained as follows:

True Positive (TP) = 145 users were correctly predicted as depressed by the model

True Negative (TN) = 206 users were correctly predicted as Non-depressed by the model

False Positive (FP) = 8 Users were incorrectly predicted as Depressed by the model

False Negative (FN) = 21 Users were incorrectly predicted as Non-depressed by the model

Table 62 Showing Results for Random Forest for user level

Measure	Value
Sensitivity (recall)	87.35%
Specificity	96.26%
Precision	94.77%
Negative predictive value	90.75%
False positive rate	3.74%
False discovery rate	5.23%
False negative rate	12.65%
Accuracy	92.37%
F1 score	90.91%
Matthews correlation coefficient	84.56%

Table 62 Shows the results for Random Forest at User Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Recall (Sensitivity) = 87.35% (represents the fraction of users detected correctly that actually represents depressed class)

Specificity = 96.26% (represents the fraction of users detected correctly that actually represents non-depressed class)

Precision = 94.77% (represents the fraction of possible predicted outcomes that were significant)

Negative Predictive Value = 90.75% (represents the correctness of probability towards the outcome of predicted “non-depressed” users that are actually “non-depressed” in real)

False Positive Rate = 3.74% (represents the percentage of users who are “non-Depressed” but identified as “depressed” users)

False Discovery Rate = 5.23% (represents the percentage of users identified as Depressed users that are actually “Non-Depressed” in real)

False Negative Rate = 12.65% (represents the percentage of users that are “Depressed” but identified as “non-depressed” user)

Accuracy = 92.37% (represents the total accuracy of prediction performance by the model)

F1 score in decimal value = 0.9091 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

Matthews correlation coefficient = 0.8456 (MCC represents the quality scores for binary classifications between the coefficient value ranges -1 to 1, where coefficient of -1 is wrong classification and coefficient of +1 is perfect classification. It helps to understand the relation between the real values and the predicted outcomes. The higher the value of MCC, the more is the quality of predictive model.

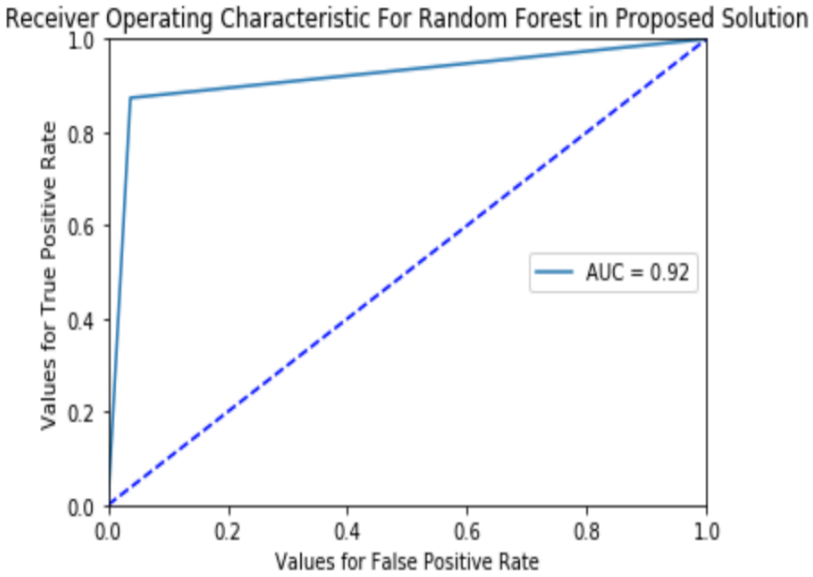


Illustration 73 Showing ROC and AUC for random forest in proposed solution

The Illustration 73 Shows ROC curve for Random Forest at User Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.92, which is above the range 0.8 to 0.9 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

6.3.2 Evaluation of Confusion Matrix for Logistic Regression for user level

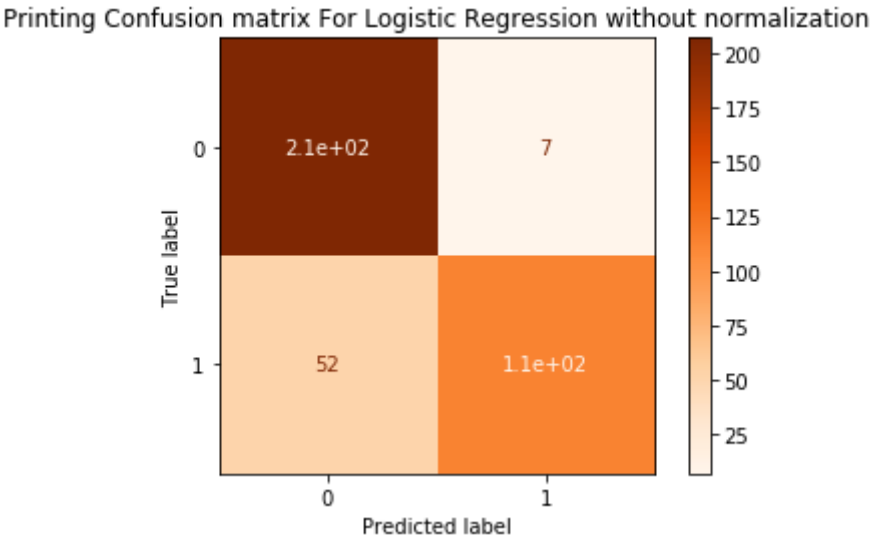


Illustration 74 Showing Confusion Matrix without normalization for Logistic Regression at user level

The illustration 74 shows a confusion matrix without normalization for Logistic Regression at User Level. Each value in the matrix shows the total number of instances in the testing data in the form of an equation. In order to view the exact value of each predicted instance the equation gets converted into numbers printed by the classifier as shown in Table 63

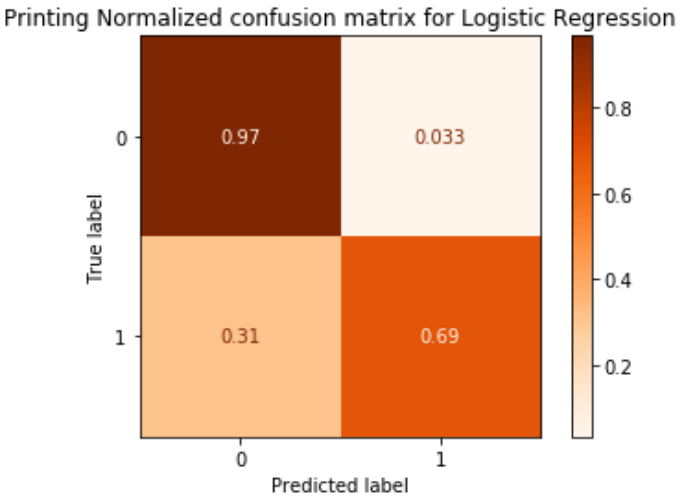


Illustration 75 Showing Normalized Confusion Matrix for Logistic Regression for user level

The Illustration 75 shows a normalized confusion matrix for Logistic Regression at User Level. The decimal representation helps to understand the results by observation of the diagonal values showing the degree of accurately predicted occurrences, which are higher than the off diagonal values. Moreover, the sum of values of each row is equal to 1.00 that shows the 100% utility of possible outcomes for each class.

Table 63 Showing confusion matrix for Logistic Regression for user level

	Depressed	Non-Depressed
Predicted as Depressed	TP = 114	FP = 7
Predicted as Non-Depressed	FN = 52	TN = 207

The Table 63 Shows the values of all possible predicted outcomes for Logistic Regression at User Level. The values in the matrix are explained as follows:

True Positive (TP) = 114 users were correctly predicted as depressed by the model

True Negative (TN) = 207 users were correctly predicted as Non-depressed by the model

False Positive (FP) = 7 Users were incorrectly predicted as Depressed by the model

False Negative (FN) = 52 Users were incorrectly predicted as Non-depressed by the model

Table 64 showing metrics results for Logistic Regression for user level

Measure	Value
Sensitivity (recall)	68.67%
Specificity	96.73%
Precision	94.21%
Negative predictive value	79.92%
False positive rate	3.27%
False discovery rate	5.79%
False negative rate	31.33%
Accuracy	84.47%
F1 score	79.44%
Matthews correlation coefficient	69.63%

Table 64 Shows the results for Logistic Regression at User Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Recall (Sensitivity) = 68.67% (represents the fraction of users detected correctly that actually represents depressed class)

Specificity = 96.73% (represents the fraction of users detected correctly that actually represents non-depressed class)

Precision = 94.21% (represents the fraction of possible predicted outcomes that were significant)

Negative Predictive Value = 79.92% (represents the correctness of probability towards the outcome of predicted “non-depressed” users that are actually “non-depressed” in real)

False Positive Rate = 3.27% (represents the percentage of users who are "non-Depressed" but identified as "depressed" users)

False Discovery Rate = 5.79% (represents the percentage of users identified as Depressed users that are actually "Non-Depressed" in real)

False Negative Rate = 31.33% (represents the percentage of users that are "Depressed" but identified as "non-depressed" user)

Accuracy = 84.47% (represents the total accuracy of prediction performance by the model)

F1 score in decimal value = 0.7944 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

Matthews correlation coefficient = 0.6963 (MCC represents the quality scores for binary classifications between the coefficient value ranges -1 to 1, where coefficient of -1 is wrong classification and coefficient of +1 is perfect classification. It helps to understand the relation between the real values and the predicted outcomes. The higher the value of MCC, the more is the quality of predictive model.

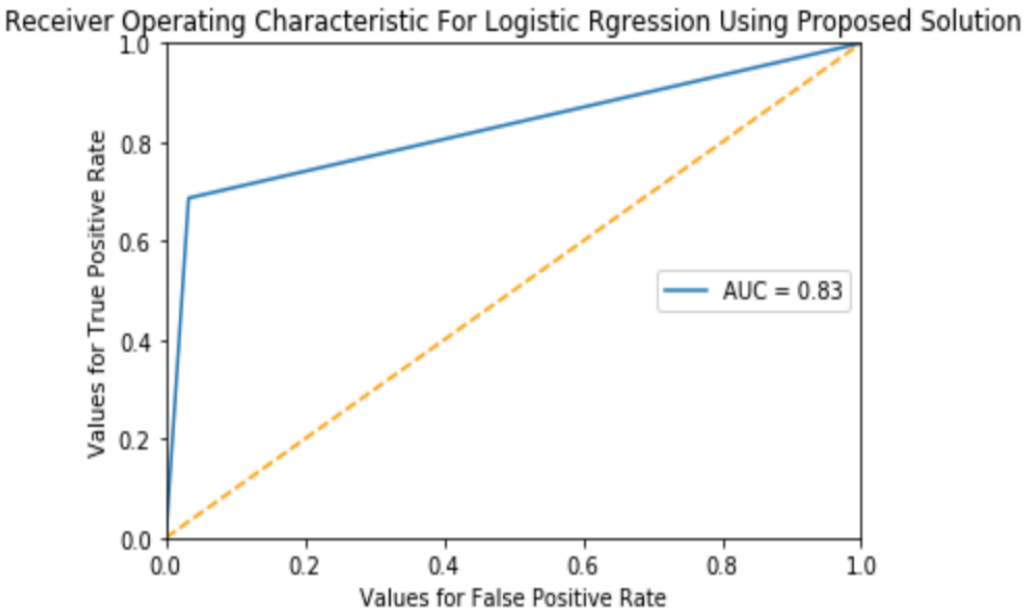


Illustration 76 Showing ROC and AUC for logistic regression in proposed solution

The Illustration 76 Shows ROC curve for Logistic Regression at User Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.83, which lies between the range 0.8 to 0.9 is considered an excellent classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

6.3.3 Evaluation of Confusion Matrix for XGBoost with Random Forest Classifier as weak learner at User Level

```
Confusion Matrix obtained for XG Boost is
 [[197  17]
 [ 12 154]]
Accuracy after applying XG Boost 0.9236842105263158
Scores after applying 2 CV on XG Boost 0.7616752675476122
Precision for XG Boost  0.9005847953216374
Recall for XG Boost  0.927710843373494
```

Illustration 77 Showing Results for XGBoost at user level

The Illustration 77 shows a screenshot of the results for XGBoost with Random Forest Classifier as weak learner at User Level. The values are represented in table 65 for better understanding of the observed and predicted outcomes.

Table 65 Confusion matrix for XGBoost for user level

	Depressed	Non-Depressed
Predicted as Depressed	TP = 154	FP = 17
Predicted as Non-Depressed	FN = 12	TN = 197

The Table 65 Shows the values of all possible predicted outcomes for XGBoost with Random Forest Classifier as weak learner at User Level. The values in the matrix are explained as follows:

True Positive (TP) = 154 users were correctly predicted as depressed by the model

True Negative (TN) = 197 users were correctly predicted as Non-depressed by the model

False Positive (FP) = 17 Users were incorrectly predicted as Depressed by the model

False Negative (FN) = 12 Users were incorrectly predicted as Non-depressed by the model

Table 66 Showing Results for XG Boost for User Level

Measure	Value
Sensitivity(recall)	92.77%
Specificity	92.06%
Precision	90.06%
Negative predictive value	94.26%
False positive rate	7.94%
False discovery rate	9.94%
False negative rate	7.23%
Accuracy	92.37%
F1 score	91.39%
Matthews correlation coefficient	84.57%

Table 66 Shows the results for XGBoost with Random Forest Classifier as weak learner at User Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Recall (Sensitivity) = 92.77% (represents the fraction of users detected correctly that actually represents depressed class)

Specificity = 92.06% (represents the fraction of users detected correctly that actually represents non-depressed class)

Precision = 90.06% (represents the fraction of possible predicted outcomes that were significant)

Negative Predictive Value = 94.26% (represents the correctness of probability towards the outcome of predicted “non-depressed” users that are actually “non-depressed” in real)

False Positive Rate = 7.94% (represents the percentage of users who are "non-Depressed" but identified as "depressed" users)

False Discovery Rate = 9.94% (represents the percentage of users identified as Depressed users that are actually "Non-Depressed" in real)

False Negative Rate = 7.23% (represents the percentage of users that are "Depressed" but identified as "non-depressed" user)

Accuracy = 92.37% (represents the total accuracy of prediction performance by the model)

F1 score in decimal value = 0.9139 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

Matthews correlation coefficient = 0.8457 (MCC represents the quality scores for binary classifications between the coefficient value ranges -1 to 1, where coefficient of -1 is wrong classification and coefficient of +1 is perfect classification. It helps to understand the relation between the real values and the predicted outcomes. The higher the value of MCC, the more is the quality of predictive model.

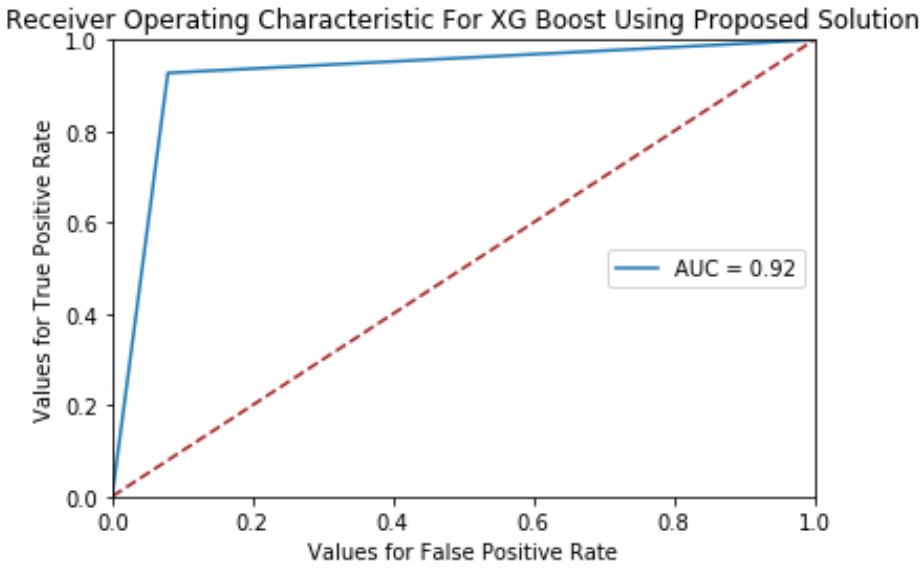


Illustration 78 Showing Roc and AUC for XGBoost for user level

The Illustration 78 Shows ROC curve for XGBoost with Random Forest Classifier as weak learner at User Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.92, which is above the range 0.8 to 0.9 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

6.3.4 Evaluation of Confusion Matrix for AdaBoost with Random Forest Classifier as weak learner at User Level

```
Finished in ...  0.5814299583435059
Confusion Matrix obtained for Adaboost is
 [[207   7]
 [ 22 144]]
Accuracy after applying Adaboost 0.9236842105263158
Scores after applying 10 CV on Adaboost 0.9394736842105263
Precision for AdaBoost  0.9536423841059603
Recall for AdaBoost  0.8674698795180723
```

Illustration 79 Showing Results for AdaBoost for user level

The Illustration 79 shows a screenshot of the results for AdaBoost with Random Forest Classifier as weak learner at User Level. The values are represented in table 67 for better understanding of the observed and predicted outcomes.

Table 67 Confusion Matrix for AdaBoost for user level

	Depressed	Non-Depressed
Predicted as Depressed	TP = 144	FP = 7
Predicted as Non-Depressed	FN = 22	TN = 207

The Table 67 Shows the values of all possible predicted outcomes for AdaBoost with Random Forest Classifier as weak learner at User Level. The values in the matrix are explained as follows:

True Positive (TP) = 144 users were correctly predicted as depressed by the model

True Negative (TN) = 207 users were correctly predicted as Non-depressed by the model

False Positive (FP) = 7 Users were incorrectly predicted as Depressed by the model

False Negative (FN) = 22 Users were incorrectly predicted as Non-depressed by the model

Table 68 Results for AdaBoost for user level

Measure	Value
Sensitivity(recall)	86.75%
Specificity	96.73%
Precision	95.36%
Negative predictive value	90.39%
False positive rate	3.27%
False discovery rate	4.64%
False negative rate	13.25%
Accuracy	92.37%
F1 score	90.85%
Matthews correlation coefficient	84.61%

Table 68 Shows the results for AdaBoost with Random Forest Classifier as weak learner at User Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Recall (Sensitivity) = 86.75% (represents the fraction of users detected correctly that actually represents depressed class)

Specificity = 96.73% (represents the fraction of users detected correctly that actually represents non-depressed class)

Precision = 95.36% (represents the fraction of possible predicted outcomes that were significant)

Negative Predictive Value = 90.39% (represents the correctness of probability towards the outcome of predicted “non-depressed” users that are actually “non-depressed” in real)

False Positive Rate = 3.27% (represents the percentage of users who are “non-Depressed” but identified as “depressed” users)

False Discovery Rate = 4.64% (represents the percentage of users identified as Depressed users that are actually “Non-Depressed” in real)

False Negative Rate = 13.25% (represents the percentage of users that are “Depressed” but identified as “non-depressed” user)

Accuracy = 92.37% (represents the total accuracy of prediction performance by the model)

F1 score in decimal value = 0.9085 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

Matthews correlation coefficient = 0.8461 (MCC represents the quality scores for binary classifications between the coefficient value ranges -1 to 1, where coefficient of -1 is wrong classification and coefficient of +1 is perfect classification. It helps to understand the relation between the real values and the predicted outcomes. The higher the value of MCC, the more is the quality of predictive model.

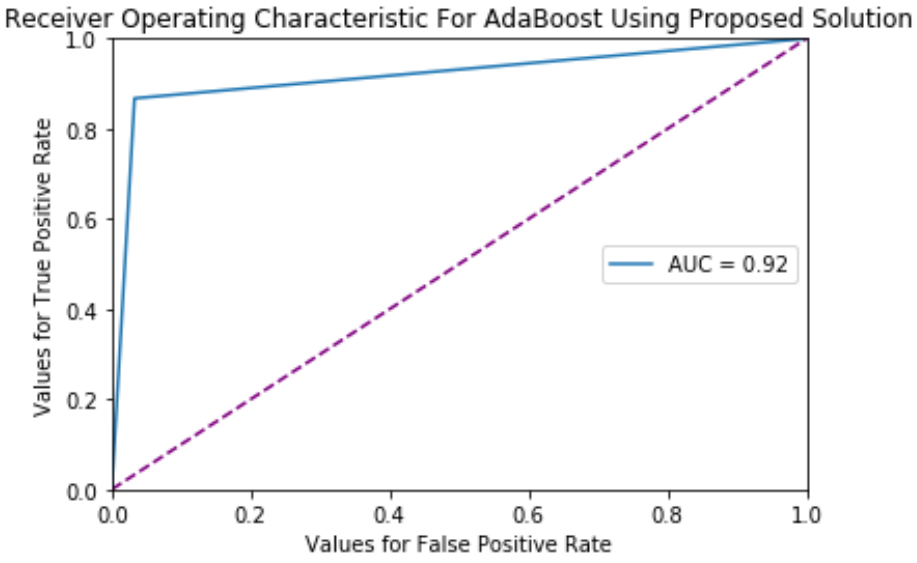


Illustration 80 Showing Roc and AUC for AdaBoost for user level

The Illustration 80 Shows ROC curve for AdaBoost with Random Forest Classifier as weak learner at User Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.92, which is above the range 0.8 to 0.9 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

6.3.5 Evaluation of Confusion Matrix for Gradient Boosting on default decision trees as weak learner at User Level

```
Finished in ...  1.1011059284210205
Confusion Matrix obtained for Gradient Boosting is
 [[206   8]
 [ 10 156]]
Accuracy after applying Gradient Boosting 0.9526315789473684
Scores after applying 10 CV on Gradient Boosting 0.9421052631578947
Precision for Gradient Boosting  0.9536423841059603
Recall for Gradient Boosting  0.8674698795180723
```

Illustration 81 Showing Results for Gradient Boost for user level

The Illustration 81 shows a screenshot of the results for Gradient Boosting on default decision trees as weak learner at User Level. The values are represented in table 69 for better understanding of the observed and predicted outcomes.

Table 69 Showing confusion matrix for Gradient Boosting for user level

	Depressed	Non-Depressed
Predicted as Depressed	TP = 156	FP = 8
Predicted as Non-Depressed	FN = 10	TN = 206

The Table 69 Shows the values of all possible predicted outcomes for Gradient Boosting on default decision trees as weak learner at User Level. The values in the matrix are explained as follows:

True Positive (TP) = 156 users were correctly predicted as depressed by the model

True Negative (TN) = 206 users were correctly predicted as Non-depressed by the model

False Positive (FP) = 8 Users were incorrectly predicted as Depressed by the model

False Negative (FN) = 10 Users were incorrectly predicted as Non-depressed by the model

Table 70 Showing Results for Gradient Boosting for user level

Measure	Value
Sensitivity(recall)	86.75%
Specificity	96.26%
Precision	95.36%
Negative predictive value	95.37%
False positive rate	3.74%
False discovery rate	4.88%
False negative rate	6.02%
Accuracy	95.26%
F1 score	94.55%
Matthews correlation coefficient	90.36%

Table 70 Shows the results for Gradient Boosting on default decision trees as weak learner at User Level. The results are represented in percentage format using various metrics that helps to compare the performance results with other models.

Recall (Sensitivity) = 86.75% (represents the fraction of users detected correctly that actually represents depressed class)

Specificity = 96.26% (represents the fraction of users detected correctly that actually represents non-depressed class)

Precision = 95.36% (represents the fraction of possible predicted outcomes that were significant)

Negative Predictive Value = 95.37% (represents the correctness of probability towards the outcome of predicted “non-depressed” users that are actually “non-depressed” in real)

False Positive Rate = 3.74% (represents the percentage of users who are "non-Depressed" but identified as "depressed" users)

False Discovery Rate = 4.88% (represents the percentage of users identified as Depressed users that are actually "Non-Depressed" in real)

False Negative Rate = 6.02% (represents the percentage of users that are "Depressed" but identified as "non-depressed" user)

Accuracy = 95.26% (represents the total accuracy of prediction performance by the model)

F1 score in decimal value = 0.9455 (represents the value for harmonic average of recall and precision. To add to it, the decimal value near to 1 indicates a perfect F1 score having balanced high precision and high recall, where a value 0 means worst F1 score having low precision and low recall)

Matthews correlation coefficient = 0.9036 (MCC represents the quality scores for binary classifications between the coefficient value ranges -1 to 1, where coefficient of -1 is wrong classification and coefficient of +1 is perfect classification. It helps to understand the relation between the real values and the predicted outcomes. The higher the value of MCC, the more is the quality of predictive model.

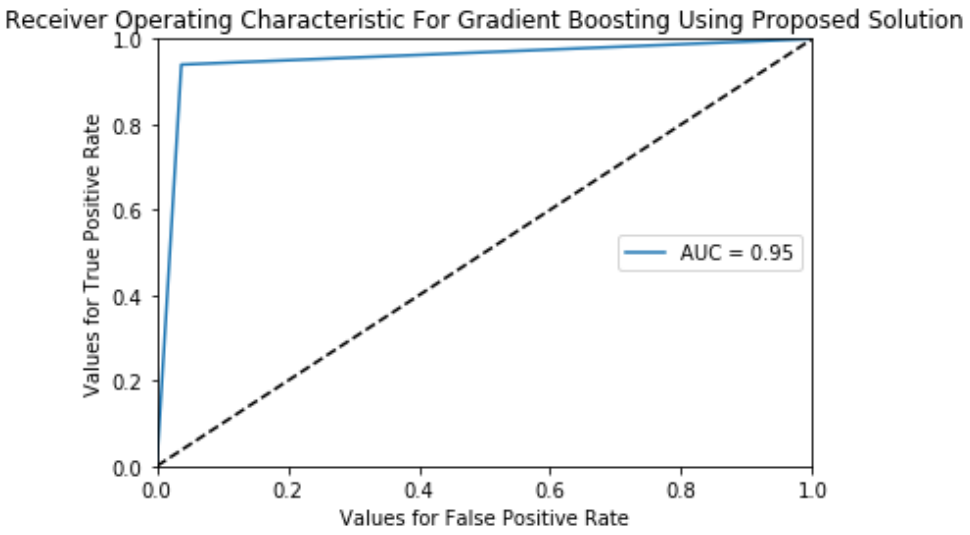


Illustration 82 Showing ROC and AUC for Gradient Boosting for user level

The Illustration 82 Shows ROC curve for Gradient Boosting on default decision trees as weak learner at User Level, depicting the probabilities of correctly classified values, and the parameter area under the ROC Curve (**AUC**) denotes how accurately the model's diagnostic ability has performed on the problem of separation. Here, the value of AUC is 0.95, which is above the range 0.8 to 0.9 is considered an outstanding classifier [106]. The higher the value of area under the curve, more is the model accurate in distinguishing between different classes.

Chapter 7: Comparison of Evaluation and Discussion

This chapter involves the comparative analysis of each of the metrics involved in experimental as well as proposed solution. Further, this chapter shows the comparative evaluation of results from each of the models used under experiments using combination of TF-IDF and Word2Vec, as well as models using combination of GloVe and TF-IDF. To add to it, a comparison using precision, recall, accuracy, F1-score, area under the curve, time complexity, K-fold cross validation scores as well as Big O notation is also performed on results from both the experimental and the proposed solution.

7.1 Comparison of Results of Previous works and Our Experiments under Tweet Level Architecture

The table 71 shows a comparison of previous works done for a tweet level architecture and results from our experiment models having highest accuracy using Word2Vec and TF-IDF under tweet level architecture. Further, the highest accuracies shown by our models are taken by SVM with 98.14% and the logistic regression with an accuracy of 97.03%.

Table 71 Showing comparison with past research papers under tweet level

COMPARISON OF OTHER MODELS FOR TWEET TO TWEET LEVEL		
ALGORITHMS	ACCURACY OF PREVIOUS MODELS AT TWEET LEVEL	ACCURACY OF OUR EXPERIMENTS AT TWEET LEVEL
RANDOM FOREST	85% [64]	95.65%
SVM	71% [18]	**98.14%**
LOGISTIC REGRESSION	92% [93]	**97.03%**
XGBOOST	N/A	95.65%
ADABOOST	79% [64]	94.58%
GRADIENT BOOST	N/A	95.50%
Bi-LSTM	80.5% [63]	94.5%

7.2 Comparison of Proposed Solution Models for User to User Level

The table 72 shows the results from all the models implemented in our User to User level model by using various metrics such as Precision, Recall, F1-Score and accuracy, Where the highest results for accuracy are achieved by gradient boosting as 95.26% and the fastest running time is shown by logistic regression with 0.0589 seconds.

Table 72 Showing comparison using various metrics for user level

ALGORITHMS	METRICS				
	PRECISION	RECALL	F1-SCORE	ACCURACY	TIME COMPLEXITY
RANDOM FOREST	94.77%	87.35%	90.91%	92.37%	0.2653 sec
LOGISTIC REGRESSION	94.21%	68.67%	79.44%	84.47%	0.0589 sec
XGBOOST	90.05%	92.77%	91.39%	92.37%	1 sec
GRADIENT BOOST	95.36%	86.75%	94.55%	95.26%	1.10 sec
ADABOOST	95.36%	86.75%	90.85%	92.36%	0.581 sec

7.3 Run Time Complexity

The Run time complexity helps us to compute the efficiency and performance of an algorithm on a time scale as well as utilizing the Big O notation for each algorithm. To calculate the time, a library having a time function is called using “Import time” and then the “time” function is operated. The pseudo code for time function is represented as (start_time=time.time()). However, the concept of Big O notation works on the concept of providing information regarding the speed of an algorithm. The outcome focus on a logic that the smaller number of loops to be declared in an algorithm shows a higher quality in

terms of time performance. Similarly, the more the number of loops in a model, the lesser is the performance in context to time scale. This is because the denotation $O(an\ Log(n)^p)$ with the (p) number of loops, records equal to (a) and attributes taken as (n) can be represented in such a manner that the more value of loops will ultimately be the dominant element in manipulating the performance of a function.

The table 73 represents the run time complexity for each of the algorithms under the user level processing 2,99,539 Tweets for 760 users with a split ratio of 50:50 for testing and training. However, the fastest run time is recorded as 0.0589 sec by logistic regression whereas the gradient boost takes 1.10 sec for processing the output.

Table 73 Showing Time Complexity of User Level Architecture

ALGORITHMS	TIME COMPLEXITY	BIG O NOTATION
RANDOM FOREST	0.2653 sec	$O(n\ Log(n_{Trees}))$
LOGISTIC REGRESSION	0.0589 sec	$O(an\ Log(n)^p)$
XGBOOST	1 sec	$O(n\ Log(n))$
GRADIENT BOOST	1.10 sec	$O(a\ Log(n)^p)$
ADABOOST	0.581 sec	$O(\ Log\ n)$

The Table 74 represents the run time complexity for each algorithm under the tweet level applied using GloVe + TF-IDF and Word2vec +TF-IDF. The fastest run time is shown by Logistic regression under Word2vec +TF-IDF with 5 seconds whereas, the maximum time to run was taken by XGBoost with random forest as a weak learner under Glove +TF-IDF for10,861 seconds.

Table 74 Showing Time Complexity of Tweet Level Architecture

ALGORITHMS	TIME COMPLEXITY		BIG O NOTATION
	GLOVE + TF-IDF	WORD2VEC +TF-IDF	
RANDOM FOREST	750 sec	636 sec	$O(a\ Log(n_{Trees})^{p})$
SVM	3091 sec	220 sec	$O(an\ Log(n)^{p})$
LOGISTIC REGRESSION	6.2 sec	5 sec	O(Log(n))
XGBOOST	10,861 sec	3059 sec	$O(an\ Log(n^{\wedge p})$
ADABOOST	111.35 sec	113.4 sec	O(n Log (n))
GRADIENT BOOST	567 sec	581 sec	$O(\ n\ Log\ n^{\wedge p})$

7.4 Comparison of All Models with K-Fold Cross Validation Scores Under User Level

The Table 75 represents K-Fold cross validation scores for algorithms under user level. The cross validation using K-folds helps to verify the model by testing various samples for producing an average outcome in terms of accuracy. The K in the table shows the folds holding different section of testing samples.

Table 75 Showing comparison of K-fold cross validation scores for user level

ALGORITHMS	K-FOLD CROSS VALIDATION SCORES	ACCURACY OF MODELS FOR USER TO USER LEVEL
RANDOM FOREST	92.36% (K=2)	92.36%
LOGISTIC REGRESSION	87.63% (K=2)	84.47%
XGBOOST	76.16% (K=2)	92.37%
ADABOOST	93.94% (K=10)	92.36%
GRADIENT BOOST	94.21% (K=10)	95.26%

7.5 Comparing Results of our proposed solution with previous works For User Level Module

The table 76 shows a comparison between results from our proposed solution and the existing research papers under a user level architecture on the metrics such as precision, recall and accuracy. The gradient boosting on default decision trees as weak learner shows the highest accuracy of 95.36% with a recall rate of 86.75% and a precision of 95.26% ultimately making it the best model.

Table 76 Showing Comparison of Results with Previous Research Papers for User level

COMPARISON OF RESULTS WITH PREVIOUS RESEARCH PAPERS	METRICS		
	PRECISION	RECALL	ACCURACY
PROPOSED SOLUTION (Gradient Boosting)	95.36%	86.75%	95.26%
Research paper [1] (SVM)	74.2%	62.9%	70.35%
PROPOSED SOLUTION (Gradient Boosting)	95.36%	86.75%	95.26%
Research paper [2] (SVM)	82.8%	67.5%	73.57%
PROPOSED SOLUTION (Gradient Boosting)	95.36%	86.75%	95.26%
Research paper [5] (SVM)	72.9%	74%	71.11%
PROPOSED SOLUTION (Gradient Boosting)	95.36%	86.75%	95.26%
Research paper [6] (Perceptron)	86.9%	74%	78.4%
PROPOSED SOLUTION (Gradient Boosting)	95.36%	86.75%	95.26%
Research paper [19] (SVM)	70.83%	85%	78.72%

7.6 Discussion - Solving Research Questions

The research questions that can be answered from this research are :

Q1. How can we make a machine learning based model that can automatically process and predict depression from different tweets?

Answer. There are two architectures for which depression gets predicted from tweets described as follows:

1. Tweet Level: The first method works with the Tweet to Tweet Level model under which Tweets can be extracted from Twitter using Tweepy (python library). After collecting the data, the cleaning of data will remove the unwanted content such as stop words, URL, punctuation marks, duplicate tweets and duplicate users. After the data cleaning Tweets can be converted in the vector format using GloVe with TF-IDF, and also with Word2Vec and TF-IDF. The converted data into vectors then will be used for training and testing events with ratio of 70:30 respectively loaded to the classifiers and the outcomes will be retrieved in terms of a confusion matrix. The confusion matrix will be used to represent the outcomes in terms of various metrics such as accuracy, precision, recall and other values where recall rate is the most important factor which represents the number of accurately predicted instances of desired class. Such classification will help to provide a technique that can predict depression from tweets connected to a user. Ultimately, providing a practical implementation of a model that can predict depression covering a large population of users on social media.

2. User Level: The Second method works with User to User Level model where the tweets within a certain timeline depending upon each depressed and non-depressed user are extracted. These Tweets are further operated for understanding the behaviour of each

user on social media using extensive feature engineering. This approach will help us to understand the pattern of depressed and non-depressed users in terms of attributes such as Extreme depress status, Night Status, Polarity contrast, Negative Polarity, Lexical Richness, User Mentions, Social Response ratio, Intensity of depression related words, Lexical Features of vocabulary. Additionally, many features can also be derived as stated in the proposed solution to convert tweets into numbers which helps us to detect depression even with a dataset where a self-declaration of user having depression is not posted. The concept of converting emoticons into text using Emojipedia [80][81] is one of the essential elements that helped to better understand the emotion from each Tweet. After the data cleaning process, the text under the tweets along with the text from emoticons gets converted into vectors using concepts under Part of Speech tagging and the outputs from each of the Extensive feature engineering attributes. The data is then fed to the classifier models for training and testing ratio of 50:50 for each class and the outcomes will be retrieved in terms of a confusion matrix. The confusion matrix will be used to represent the outcomes in terms of various metrics such as accuracy, precision, recall and other values where recall rate is the most important factor which represents the number of accurately predicted instances of desired class. Such classification model will help to predict a depressed user by learning the pattern of social media usage and attributes operated under extensive feature engineering from actually depressed users on social media who have made self-declaration posts for being diagnosed with depression. The results from the gradient boosting with a high recall value ultimately makes this model practical to work with random nature of real time tweets.

Q2. What is the relation between different types of attributes involved in the dataset constructed using the real-time extraction of tweets?

Answer. The real-time data in most cases need extensive data cleaning which allows feature engineering concepts to create serval variations required for a particular type of data. These variations represent certain content in terms of feature variables (attributes). Further, to organize such feature variables for a large set of data to interpret more advanced statistics a correlation matrix is formed. The correlation matrix helps to represent the data in terms of feature variables (attributes) to understand the relationship between each feature in terms of a correlation coefficient value. The correlation matrix in illustration 70, represents a table showing an observable pattern of dark red to blue colour boxes with numbers, where the numbers are the correlation coefficient values between any two variables. The diagonal value is set to 1 because it shows the bond between a same variable with itself. However, according to the correlation matrix, there are some connections between different types of feature variables (attributes). The attribute extreme depresses status and night status, share a strong correlation indicating that the users tend to post negative tweets that contain depression related words during night hours. Also, there is weak correlation between the lexical richness of sentence if posted during night hours. The lexical features such as adjective and adverbs share a strong correlation.

Some of other correlation coefficient values between various attributes from high to low bonding are as following:

1. Adjective and adverb with correlation coefficient of (0.83). It represents a strong connection between the posts where adjectives as well as adverbs are interlinked to

detect the sentence construction of sentences to distinguish between posts by depressed and non-depressed users.

2. User mentions per tweet and all user mentions with correlation coefficient of (0.77). It represents of social involvement of user on Twitter. The number of users mentioned by a user and the number of total users mentions in their whole timeline of 1 month of tweets are interlinked with a high value, which indicates to be a good variable in distinguishing between the pattern of a depressed user with a non-depressed user in terms of the user mentions in their posts.
3. Adjectives and noun with correlation coefficient of (0.61). It represents a relation under which the ability of a user to use adjectives and noun are analyzed in distinguishing between how a user refer to different entities or objects as well as how the adjectives are used in their posts.
4. Night status and extreme depress status with correlation coefficient of (0.56). It represents a relation that the people who posts more during the night hours tend to use more negative words.
5. User response ratio and user response ratio per tweet with correlation coefficient of (0.47). It shows the relation between user's reply, retweet and like count for each tweet and the response ratio for the whole tweets over a month. Ultimately forming a pattern between both attributes to distinguish between the depressed and the Non-depressed users.
6. Negative polarity of tweets and extreme depress status with correlation coefficient of (0.31). It shows that the users who have posted more than 20% of the posts expressing

negative emotions showing higher negative polarity are interlinked with using more depressive words under extreme depress status attribute to some extent.

7. Negative polarity of tweets and night status with correlation coefficient of (0.29). It shows that the people who posts during the night hours have relationship with the negative polarity of tweets in context with the use of negative words detected using TextBlob [90]
8. Lexical richness and extreme depress status with correlation coefficient of (-0.41). It represents a negative value which shows that the users with extreme depress status attribute have a very low use of vocabulary and unique words. Ultimately, decreasing the correlation coefficient value for lexical richness and extreme depress status.

Q3. What type of characteristics we can extract from tweets to classify users into the categories of depressed and non-depressed user ?

Answer. There are various factors that play an important role in determining the depression detection that can be obtained from the user profile, out of which some of the key factors include person's following count, person's followers count, user mentions, as well as social response ratio and others. The social behaviour of the user while interacting with other users in terms of tweets and replies or even the like and follow counts provide an overview for distinguishing one user to another. In other words, a depressed user will represent a different pattern as compared to a non-depressed user in terms of social media statistics. Some of the factors are described as following:

1. We can extract the text from tweets of users and convert it into vectors in terms of Adverbs , adjectives, nouns, as well as proper nouns which will provide outputs in terms

of Lexical Features of Vocabulary with respect to each user's sentence formation helps in classifying the categories for depressed and non-depressed users.

2. the engagement of a user on social media during night hours with the use of negative words can be detected by using night status as well as applying the extreme depress status attribute.

3. The tweets can be classified further into positive and negative sentiments by using emojipedia that helps to convert the emoticons in the tweets into their emotional meanings and then by applying Textblob[90] that ultimately calculates the sentiment of tweets.

4. Features such as Polarity contrast and negative polarity of tweets helps to classify the words into a vector which matches the degree of mood shifts in sentences where a user is posting a tweet with positive words and after a few hours of the tweet the same user is posting tweets with negative words. For an example if a user posts at 7pm " Enjoying with friends" and the same user after few hours posts "Not able to sleep, feeling alone hoping this life to end". This situation highlights a shift in the mood for a user which can be captured by using Polarity contrast feature.

Furthermore, by implementing all these factors in terms of training and testing data the tweets can be classified into depressed and non-depressed class.

Q4. What kinds of analytical facts can be recognized for users in different age groups on social media platform (Twitter)?

Answer. The age of users was calculated and added as a new feature as a part of research analysis. As per the Twitter user registration norms, the user can register himself or

herself for account creation at the age of 13 [94]. Based on the hypothesis, that the person started using twitter at the age of 16, or the user has created the Twitter account at the age of 16, the value of the age attribute of each user is inferred. For instance, if the user has created the account in year 2013, then its assumed age will be (present year – the account creation year +16 [default age assumed]), which is, 2020 – 2013 +16 = 23. However, the construction of age attribute based on the real hypothesis unveils several facts according to the dataset for this research as following:

1. Number of Users by Age: The maximum number of users recorded as 15256 belongs to the age group of 27, Whereas the users with age 25, recorded as 12012 in number are second highest present in the dataset. The lowest number of users recorded as 34 belongs to the age group 30.
2. User category by Age: 11,400 users belonging to the age group 27 falls under the category of non-depressed users. Whereas, 3856 users belonging to same age group of 27 falls under the category of depressed users. However, nearly 11,000 users who belong to depressed category falls under the age group of 18,19, 24, and 25.
3. Correlation between Age and hours of Tweet Post: The illustration 53 records the relationship between age and time-series statistics of all the posts created during a day by each user. For instance, 900 users within the age group of 27 years posted their tweets between 2am to 5am and 9pm to 10pm. Similarly, for age group 24 and 25, nearly 600 people posted their tweets between 12am to 3am and 10pm to 11pm. However, the busiest hours during the morning time can be observed near 3am when the depressed and the non-depressed users were online.

Q5. What other demographic attributes influence the detection of depression on social media?

Answer. Various Analytical factors could manipulate the accuracy of the model in order to predict depression on social media such as location and type of data source, dataset size, feature engineering criteria as well as selecting Classification models .The location of the Tweets for extraction should cover a larger area and not focus on a particular region or state. The reason behind selecting a vast area is that the problems related with mental health might be vary from place to place and people belonging to several countries. For instance, the people in countries such as US and Canada are more open to share their thoughts related to mental health as compared to people in Japan. Similarly, Canadian campaign "Bell Let's Talk" [97] changes the whole scope for generating new information and content related with mental illness by allowing people to post about their mental health problems using #BellLetsTalk [97] on social media. Moreover, a higher self-disclosure rate helps to increase the available informative content for real time extraction of data on social media. To conclude, for generalising a model on a larger scale the selection of location should not be fixed to a particular place. The data set size should be taken in such a manner that the number of depressed as well as the non-depressed users counterbalance the predictable instances within a specific time period to avoid the problems of overfitting and underfitting. For instance, if we extract 2 months of data for a depressed user and for a non-depressed user, there might be a bias in the collection of data because the time period for informative tweets for a depressed user might not be same for a non-depressed user. Further, selection of a high-performance classification model is important because in order to decrease the error rate of the model,

the recall value should be higher for the outcomes of prediction by the classifier. Some of the factors that influences the predictive outcomes can be splitting the data into appropriate testing and training ratio, applying extensive data cleaning in order to remove duplicate records in data as well as application of a cross validation method such as K-fold test score that helps to test the model with K samples for complete verified results. Moreover, some of the other demographic attributes include linguistic and semantic features along with the formation of sentences by the users, discussed under feature engineering can also influence the results. However, the night status as well as the extreme depress status have a greater impact on detection of depression.

Q6. What types of patterns in vocabulary and sentence construction can be observed from users on social media.?

Answer. The correlation matrix also uncovers the fact that the sentence construction capability during nighttime is low for depressed users. However as compared by the use of internet at night, the depressed users are more actively online during night hours on social media as compared to the non-depressed users. Further, combining the data for training dataset the users for both the depressed and non-depressed categories were classified under lexical features of vocabulary, where out of 760 users the Lexical richness Attribute ratio on a scale of 0 to 1 was taken. The lexical richness can be defined as the ratio of unique words in the sentence to the total words in the sentence. The ratio above 0.8 was shown by nearly 20 users whereas the lexical ratio above 0.2 was observed for nearly 200 users and the remaining users were between the lexical ratio of 0.0 to 0.2. Further, the ratio of proper noun per sentence was taken where out of 760 users nearly

400 users were recorded as having a low ratio for implementing proper nouns in their tweets during 1-month period. On the other hand, the Pronoun ratio for 760 users show that around 100 users have higher implementation of pronouns in construction of their sentences during a period of 1 month.

Q7. How can we utilize the features such as location and user engagement data available on twitter?

Answer. The hidden features such as account creation date is extracted from the twitter timeline in order to construct the variable age. Also, collectively several activities such as likes, shares can be aggregated to analyze the user engagement of the user on social media. The response ratio attribute and the user mention attribute help to understand the importance of each element where the depressed users have made less retweets, likes and reply counts as compared to non-depressed users which can be further helpful in categorizing the outputs into vectors for analyzing a pattern to classify testing data with higher accuracy rate. The relationship between different types of features extracted from each user depends upon the quality of tweets. Twitter allows users to create a personal timeline without verification of the real identity. This allows user to add any age, location, username and other information. For instance, under such a condition if a user has entered any other imaginary location like “xyz”, the data cleaning of the extraction of tweets will result in an error while fetching and converting the location into latitude and longitude. Further, the location plays an important role for collection of tweets from all around the world in order to create a generalised dataset. The generalized dataset helps to avoid biased results that could arise if the tweets are taken from a specific place or

region where the informative content is less efficient to produce a high-quality outcome. Thus, plotting a location graph plays an important role in a model to understand the origin of the tweets and statistics related with the number of users from each location.

Q8. Which machine learning techniques work the best to detect depression in a user from their tweets on social media and how can we validate the accuracy of such prediction?

Answer. The best classifier under tweet level classification is Support vector machine (SVM) with customized trained Word2Vec embeddings and TF-IDF recorded at an accuracy of 98.14 % and recall value of 95.63%. The models can be validated by using K-Fold cross validation method where K is the number of folds for each section which holds a different sample data for testing and provides an accuracy for each test sample. Further, the average of all the test results are calculated to give a validate accuracy of the model. However, for testing the SVM model with Word2Vec + TF-IDF at Tweet Level, the value for K is set to 10 and the score was recorded at 97.80%. On the other hand, for the user level classification, the best algorithm performance is carried by Gradient Boosting with default Decision Tress as weak learner, showing an accuracy of 95.26 % and a recall value of 86.75%. The results are validated by 10-fold cross validation methods with a score of 94.21%. To conclude, the Gradient Boosting with Default Decision Trees as Weak Learner at User Level works as the best model to predict depression in user from their Tweets on social media.

Chapter 8: Conclusions and Future work

In this chapter, we present the conclusions and future work.

8.1 Conclusions

In this research, we study existing approaches for detecting and predicting depression of Twitter users. We also have compared existing solutions, and also have proposed two approaches for detecting and predicting depression among Twitter users. The first approach includes the conversion of Twitter sentences into GloVe and Word2Vec vectors and applying different classification models on them. Support Vector Machine Classifier yields the highest accuracy 98.14%, precision of 96.73% and recall rate of 95.63%, which is higher than [93], when inputted with combination of Word2Vec and TF-IDF embeddings. On the other hand, the SVM model when inputted with combination of GloVe and TF-IDF embeddings yields the accuracy 93.91% which is higher than the other models operated using GloVe and TF-IDF. The second approach for User level model utilizes extensive feature engineering to generate hidden as well given attributes other than the tweets to study the overall impact of depression. Utilizing this approach, the best performing algorithm is Gradient Boosting with default decision trees as weak learner holding an accuracy of 95.26 %, precision of 95.36% and recall rate of 86.75%. Nevertheless, the depression among users have high influence on their usage of social media, e.g., night-time usage of social media, language and vocabulary of depressed user, as well as sentiments of user also indicate the difference between depressed and non-depressed classes. Depressed users tend to share negative thoughts widely than average population and depressed users were more active during mid-morning hours from 3 am to 6 am.

Moving further, the practical application for experimental model of this research under Tweet Level can be implemented to create 50-dimensional embeddings of tweets generated by users on social media (Twitter) using combination of Word2vec and TF-IDF. Further, the outcomes generated from this combination of Word2vec and TF-IDF can be input directly into our trained experimental model for testing, to predict depression in users from their Tweets. Nevertheless, the practical application of proposed solution model of this research under User Level can be implemented in various fields by the following:

1.Big Data Analysis: This research can be utilized by other researchers to predict Depression in users on a large-scale dataset. To add to it, by using this work, other researchers can study various factors described under the Extensive feature engineering criteria under proposed solution, as well as study the pattern for engagement of different users on social media (Twitter).

2. Mental Health Professionals: The professionals such as psychologists and psychiatrists can utilize this research work to analyze their patient's activity on a social media platform(Twitter). To add to it, this research will help to categorize each of the patient using extensive feature engineering criteria on their social media posts (Tweets). Ultimately, helping to predict depression in patients and understanding the pattern of those patients that were unable to communicate their thoughts and feelings on a face to face session. Moreover, this research could also be utilized to predict depression in users by analyzing their social media (Twitter) activities at an earlier stage, that could be further given proper medical treatment to avoid severe consequences of depression (for example suicide).

8.2 Future Work

The ending of this research will delineate some of the future work techniques and concepts that will be developed as per according to the different datasets as well as combining of a more complex hybrid architecture or an AI-operated multimodal feature engineering structure. Depression detection is one of the major concerns for the proper mental health of a person and social media has empowered by becoming itself as a tool to detect depression. This research work can be extended by imputing various other features other than textual information such as profile picture of the user, sentiments of the posts shared if it contains images or videos. Also, multiple social media platforms such as Facebook, Instagram can also be exploited to generate the history of the user and also study the type of behavior exhibited by the user on various social media platforms. From the perspective of machine learning, this research work outperforms many existing approaches, but still can be enhanced by employing rigorous ensemble machine learning techniques that are built on top of other classification models to improve the accuracy. Consequently, the study of various factors accelerating the depression based on the geographic location of the user can be carried out and necessary help can be provided to the affected patient. Nevertheless, this research could also be applied to predict other mental health disorders by altering some of the data collection methods in terms of the desired outcome. For instance, Schizophrenia, Anxiety, PTSD, etc. can be predicted by modifying the phrases as well as keywords under the data extraction tools according to symptoms of mental health disorder to be analysed. This research could also be applied to study the difference between the patterns of users having different mental health disorders by collecting data for each mental health disorder in different datasets and then

passing those datasets separately into the implementation process described under the proposed solution. This will result in 2 separate outcomes from each dataset that could be investigated further. To add to it, this research could also be modified for future use where the database could be connected with an automated alert messaging system and the output received from the classification models can be selected as the target outcome which will directly alert the users in early depression stage predicted according to their social media activities. In addition to it, a dynamic website could also be designed that could connect with the techniques in this research to study and predict depression levels by simply passing the user's profile link into the website page. The interactive website could also be used for spreading awareness about various mental health disorders. The outcomes could be modified for predicting other mental health disorders such as Stress, Anxiety, Psychotic Disorders etc. by modifying the pre-processing methodology of the input dataset. This will result in enhancing the additional elements of the website where any user can create their account and input their social media details to gain information related to their mental health analyzed by the pattern of their social media activities.

Chapter 9: Bibliography

[1] Gamon, Michael & Choudhury, Munmun & Counts, Scott & Horvitz, Eric. (2013). *Predicting Depression via Social Media*. Association for the Advancement of Artificial Intelligence. URL:https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/icwsm_13.pdf

[2] Munmun De Choudhury, Scott Counts, and Eric Horvitz. 2013. *Social media as a measurement tool of depression in populations*. In Proceedings of the 5th Annual ACM Web Science Conference (WebSci '13). Association for Computing Machinery, New York, NY, USA, 47–56. DOI: https://doi.org/10.1145/2464464.2464480.

[3] Appel, Helmut, Alexander L. Gerlach, and Jan Crusius. "*The interplay between Facebook use, social comparison, envy, and depression*." Current Opinion in Psychology 9 (2016): 44-49. URL:https://www.sciencedirect.com/science/article/pii/S2352250X15002535

[4] Guntuku, Sharath Chandra, David B. Yaden, Margaret L. Kern, Lyle H. Ungar, and Johannes C. Eichstaedt. "*Detecting depression and mental illness on social media: an integrative review*." Current Opinion in Behavioral Sciences 18 (2017): 43-49. URL:https://www.sciencedirect.com/science/article/pii/S2352154617300384?dgcid=api_sd_search-api-endpoint

[5] Munmun De Choudhury, Scott Counts, and Eric Horvitz. 2013. *Predicting postpartum changes in emotion and behavior via social media*. In Proceedings

of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). Association for Computing Machinery, New York, NY, USA, 3267–3276. URL:https://dl.acm.org/citation.cfm?id=2466447

[6] Sairam Balani and Munmun De Choudhury. 2015. *Detecting and Characterizing Mental Health Related Self-Disclosure in Social Media*. In Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '15). Association for Computing Machinery, New York, NY, USA, 1373–1378. URL:https://dl.acm.org/citation.cfm?id=2732733

[7] Lydia Manikonda and Munmun De Choudhury. 2017. *Modeling and Understanding Visual Attributes of Mental Health Disclosures in Social Media*. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). Association for Computing Machinery, New York, NY, USA, 170–181. URL:https://dl.acm.org/citation.cfm?id=3025932

[8] Gkotsis, George, Anika Oellrich, Tim Hubbard, Richard Dobson, Maria Liakata, Sumithra Velupillai, and Rina Dutta. "*The language of mental health problems in social media*." In Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology, pp. 63-73. 2016. URL:https://www.aclweb.org/anthology/W16-0307.pdf

[9] Coppersmith, Glen, Kim Ngo, Ryan Leary, and Anthony Wood. "*Exploratory analysis of social media prior to a suicide attempt*." In Proceedings of the Third

Workshop on Computational Linguistics and Clinical Psychology, pp. 106-117. 2016.

URL:https://www.aclweb.org/anthology/W16-0311.pdf

[10] Sho Tsugawa, Yusuke Kikuchi, Fumio Kishino, Kosuke Nakajima, Yuichi Itoh, and Hiroyuki Ohsaki. 2015. *Recognizing Depression from Twitter Activity*. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15). Association for Computing Machinery, New York, NY, USA, 3187–3196.

URL:https://dl.acm.org/citation.cfm?id=2702280

[11] Steers, Mai-Ly N., Robert E. Wickham, and Linda K. Acitelli. "*Seeing everyone else's highlight reels: How Facebook usage is linked to depressive symptoms*." Journal of Social and Clinical Psychology 33, no. 8 (2014): 701-731.

URL:https://guilfordjournals.com/doi/pdfplus/10.1521/jscp.2014.33.8.701

[12] Verduyn, P., Lee, D. S., Park, J., Shablack, H., Orvell, A., Bayer, J., et al. (2015). *Passive Facebook usage undermines affective well-being: Experimental and longitudinal evidence*. Journal of Experimental Psychology: General, 144(2), 480-488.

DOI:http://dx.doi.org.proxy.library.carleton.ca/10.1037/xge0000057

[13] Vogel, E. A., Rose, J. P., Roberts, L. R., & Eckles, K. (2014). *Social comparison, social media, and self-esteem*. Psychology of Popular Media Culture., 3(4), 206-222.

URL: http://dx.doi.org.proxy.library.carleton.ca/10.1037/ppm0000047

[14] Who compares and despairs? The effect of social comparison orientation on social media use and its outcomes URL:https://www.sciencedirect.com/science/article/pii/S0191886915004079

[15] Appel, Helmut & Crusius, Jan & Gerlach, Alexander. (2015). *Social Comparison, Envy, and Depression on Facebook: A Study Looking at the Effects of High Comparison Standards on Depressed Individuals*. Journal of Social and Clinical Psychology. URL:https://guilfordjournals.com/doi/pdfplus/10.1521/jscp.2015.34.4.277

[16] A Shen, Guangyao, Jia Jia, Liqiang Nie, Fuli Feng, Cunjun Zhang, Tianrui Hu, Tat-Seng Chua, and Wenwu Zhu. "Depression Detection via Harvesting Social Media: A Multimodal Dictionary Learning Solution." In IJCAI, pp. 3838-3844. 2017.

[17] Jia Jia. 2018. *Mental Health Computing via Harvesting Social Media Data*. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18). URL:https://www.ijcai.org/proceedings/2018/0808.pdf.

[18] Amir Hossein Yazdavar, Hussein S. Al-Olimat, Monireh Ebrahimi, Goonmeet Bajaj, Tanvi Banerjee, Krishnaprasad Thirunarayan, Jyotishman Pathak, and Amit Sheth. 2017. *Semi-Supervised Approach to Monitoring Clinical Depressive Symptoms in Social Media*. In Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017 (ASONAM '17), Jana Diesner, Elena Ferrari, and Guandong Xu (Eds.).

ACM, New York, NY, USA, 1191-1198. DOI: https://doi.org/10.1145/3110025.3123028.

[19] Jamil, Zunaira & Inkpen, Diana & Buddhitha, Prasanth & White, Kenton. (2017). *Monitoring Tweets for Depression to Detect At-risk Users*. Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology — From Linguistic Signal to Clinical Reality, 32-40. 10.18653/v1/W17-3104. URL: "https://www.aclweb.org/anthology/W17-3104.pdf".

[20] Leis, A., Ronzano, F., Mayer, M.A., Furlong, L.I., & Sanz, F. (2019). *Detecting Signs of Depression in Tweets in Spanish: Behavioral and Linguistic Analysis*. Journal of medical Internet research, DOI: "https://doi.org/10.2196/14199".

[21] M. M. Al Darwish and H. F. Ahmad, "*Predicting Depression Levels Using Social Media Posts*," 2017 IEEE 13th International Symposium on Autonomous Decentralized System (ISADS), Bangkok, 2017, pp. 277-280. DOI: 10.1109/ISADS.2017.41, URL:https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=& arnumber=7940253.

[22] Wongkoblap, A., Vadillo, M.A., & Curcin, V. (2017). *Researching Mental Health Disorders in the Era of Social Media: Systematic Review*. Journal of medical Internet research.URL:" https://www.jmir.org/2017/6/e228/pdf".

[23] Prieto, V. M., Matos, S., Álvarez, M., Cacheda, F., & Oliveira, J. L. (2014). *Twitter: a good place to detect health conditions*. PloS one, 9(1), e86191. DOI:10.1371/journal.pone.0086191, URL:"https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3906034/pdf/pone.0086191.pdf".

[24] A.G. Reece, A.J. Reagan, K.L.M. Lix, P.S. Dodds, C.M. Danforth, E.J. Langer *Forecasting the Onset and Course of Mental Illness with Twitter Data,* (2016), URL:https://www.nature.com/articles/s41598-017-12961-9

[25] *A social media-based index of mental well-being in college campuses*, Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (2017), URL:"https://dl.acm.org/doi/10.1145/3025453.3025909".

[26] Berry, N., Lobban, F., Belousov, M., Emsley, R., Nenadic, G., & Bucci, S. (2017*). #WhyWeTweetMH: understanding why people use Twitter to discuss mental health problems*. Journal of medical Internet research, 19(4), e107. URL:"https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5399219/".

[27] Mowery, D., Smith, H., Cheney, T., Stoddard, G., Coppersmith, G., Bryan, C., & Conway, M. (2017). *Understanding depressive symptoms and psychosocial stressors on Twitter: a corpus-based study*. Journal of medical Internet research, 19(2), e48. URL:"https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5350450/".

[28] Behavioural Health of the Palm Beache*s, "Are Social Media and Depression Usage Linked? Why?"*, URL:https://www.bhpalmbeach.com/are-depression-and-social-media-usage-linked/".

[29] Lujing Chen, Web Article on "*Basic Ensemble Learning*", 2nd Jan (2019), URL:"https://towardsdatascience.com/basic-ensemble-learning-random-forest-AdaBoost-gradient-boosting-step-by-step-explained-95d49d1e2725".

[30] Joseph Rocca, "*Ensemble methods: bagging, boosting and stacking*", April 22, (2019), URL:"https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205".

[31] Stacey Ronaghan, (2018), "*The Mathematics of Decision Trees, Random Forest and Feature Importance in Scikit-learn and Spark*", URL:"https://towardsdatascience.com/the-mathematics-of-decision-trees-random-forest-and-feature-importance-in-scikit-learn-and-spark-f2861df67e3".

[32] Saimadhu Polamuri, (2017), "*How The Random Forest Algorithm Works In Machine Learning*", URL:"https://dataaspirant.com/2017/05/22/random-forest-algorithm-machine-learing/".

[33] Madhu Sanjeevi, (2017) "*Chapter 4: Decision Trees Algorithms*", URL:"https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1".

[34] Webpage "*Sentiment analysis site "Sentiment140*, URL:"http://help.sentiment140.com/other-resources".

[35] Go, Alec, Richa Bhayani, and Lei Huang. "*Twitter sentiment classification using distant supervision*." CS224N Project Report, Stanford 1.12 (2009): 2009.

[36] Twitter homepage, URL:"*Twitter. It's what's happening*".

[37] Mohammad, Saif M., and Peter D. Turney. "*Emotions evoked by common words and phrases: Using Mechanical Turk to create an emotion lexicon*." Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text. Association for Computational

Linguistics, 2010. URL:“https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm”.

[38] Yates, Andrew, Arman Cohan, and Nazli Goharian. "*Depression and self-harm risk assessment in online forums*." arXiv preprint arXiv:1709.01848, (2017), URL:“https://georgetown-ir-lab.github.io/emnlp17-depression/”.

[39] Reddit homepage, URL:“*Reddit: the front page of the internet*”.

[40] Cattell, Heather EP, and Alan D. Mead. "*The sixteen personality factor questionnaire (16PF)*." The SAGE handbook of personality theory and assessment 2 (2008): 135-178.

[41] Survey for research paper [40], *16 Personality Factor Questionnaire*, URL:“https://openpsychometrics.org/tests/16PF.php”

[42] Antony, Martin M., et al. "*Psychometric properties of the 42-item and 21-item versions of the Depression Anxiety Stress Scales in clinical groups and a community sample*." Psychological assessment 10.2 (1998): 176.

[43] Research questionnaire for [42], *Depression Anxiety Stress Scales* URL:“https://openpsychometrics.org/tests/DASS/” .

[44] *Depression Detection via Harvesting Social Media: A Multimodal Dictionary Learning Solution*, URL:“http://depressiondetection.droppages.com/”.

[45] Gratch J, Artstein R, Lucas GM, Stratou G, Scherer S, Nazarian A, Wood R, Boberg J, DeVault D, Marsella S, Traum DR. *The Distress Analysis Interview Corpus of human and computer interviews*. InLREC 2014 May (pp. 3123-3128).

[46] Enrique Garcia-Ceja, Michael Riegler, Petter Jakobsen, Jim Tørresen, Tine Nordgreen, Ketil J. Oedegaard, Ole Bernt Fasmer, Depresjon: *A Motor Activity Database of Depression Episodes in Unipolar and Bipolar Patients*, In MMSys'18 Proceedings of the 9th ACM on Multimedia Systems Conference, Amsterdam, The Netherlands, June 12 - 15, 2018.

[47] Berle, J. O., Hauge, E. R., Oedegaard, K. J., Holsten, F., & Fasmer, O. B. (2010). *Actigraphic registration of motor activity reveals a more structured behavioural pattern in schizophrenia than in major depression*. BMC research notes, 3(1), 149.

[48] *Let's Get Healthy California*, URL: "https://letsgethealthy.ca.gov/".

[49] Scrapped twitter data, Raw file webpage link URL: "https://raw.githubusercontent.com/niquejoe/Classification-of-Depression-on-Social-Media-Using-Text-Mining/master/data/tweetdata.txt".

[50] Rahul Saxena, "*HOW DECISION TREE ALGORITHM WORKS*", (2017), URL: "https://dataaspirant.com/2017/01/30/how-decision-tree-algorithm-works/".

[51] Vishal Morde, "*XGBoost Algorithm: Long May She Reign!",* (2019), URL: "https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d".

[52] Jake Hoare, "*Gradient Boosting Explained – The Coolest Kid on The Machine Learning Block*", URL:"https://www.displayr.com/gradient-boosting-the-coolest-kid-on-the-machine-learning-block/".

[53] Jason Brownlee, "*A gentle introduction to XGBoost*", (2016) URL:"https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/".

[54] Jason Brownlee "*Boosting and AdaBoost for Machine Learning*", (2016) URL:https://machinelearningmastery.com/boosting-and-AdaBoost-for-machine-learning/.

[55] Avinash Navlani, "*AdaBoost classifier in Python*", (2018) URL:https://www.datacamp.com/community/tutorials/AdaBoost-classifier-python.

[56] Michael Nielsen, "*Neural Networks and Deep Learning"*, (2019), URL:"http://neuralnetworksanddeeplearning.com/index.html".

[57] James le, "*A Gentle Introduction to Neural Networks for Machine Learning*", (2018), URL:"https://www.codementor.io/@james_aka_yale/a-gentle-introduction-to-neural-networks-for-machine-learning-hkijvz7lp".

[58] World Health Organization, "*Depression*", 30th January (2020), URL:"https://www.who.int/news-room/fact-sheets/detail/depression".

[59] Richard Gall, "*What is LSTM*?", April 11, (2018), URL:"https://hub.packtpub.com/what-is-lstm/".

[60] Web page article, ML Glossary, "*Logistic Regression*", URL:"https://ml-cheatsheet.readthedocs.io/en/latest/logistic_regression.html".

[61] Lasse Schultebraucks, "*Introduction to Support Vector Machines*", September 22, (2017), URL:"https://medium.com/@LSchultebraucks/introduction-to-support-vector-machines-9f8161ae2fcb".

[62] Pathmind, "*A Beginner's Guide to Bag of Words & TF-IDF*", URL:"https://pathmind.com/wiki/bagofwords-TF-IDF".

[63] Orabi, Ahmed Husseini, et al. "*Deep learning for depression detection of twitter users*." Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic. 2018.

[64] Tadesse, Michael M., et al. "*Detection of Depression-Related Posts in Reddit Social Media Forum*." IEEE Access 7 (2019): 44883-44893.

[65] *Twint -Twitter Intelligence Tool*, version 2.1.14, "Twintproject/Twint" , URL:"https://github.com/twintproject/twint".

[66] *Tweepy*, Version 3.8.0, URL:"http://docs.tweepy.org/en/v3.8.0/getting_started.html".

[67] Speriosu, Michael, et al. "*Twitter polarity classification with label propagation over lexical links and the follower graph*." Proceedings of the First workshop on Unsupervised Learning in NLP. Association for Computational Linguistics, 2011.

[68] Zhang, Shichao, Chengqi Zhang, and Qiang Yang. "*Data preparation for data mining*." Applied artificial intelligence 17.5-6 (2003): 375-381.

[69] Ramos, Juan. "*Using TF-IDF to determine word relevance in document queries*." Proceedings of the first instructional conference on machine learning. Vol. 242. 2003.

[70] keitakurita "*Paper Dissected: "GloVe: Global Vectors for Word Representation, Explained*", April 29, (2018),

URL:“https://mlexplained.com/2018/04/29/paper-dissected-GloVe-global-vectors-for-word-representation-explained/”.

[71] Analytics Vidhya “*An End-to-End Guide to Understand the Math behind XGBoost*” , URL : https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/

[72] Dimitris Leventis “*XGBoost Mathematics Explained*” , URL : https://towardsdatascience.com/xgboost-mathematics-explained-58262530904a

[73] Savan Patel “ *AdaBoost Classifier* “ , URL : https://medium.com/machine-learning-101/https-medium-com-savanpatel-chapter-6-AdaBoost-classifier-b945f330af06.

[74] Dmitriy Selivanov, “*GloVe Word Embeddings*”, 21 december, (2018), URL:“http://text2vec.org/GloVe.html”.

[75] Michael Nielson “ *How the background algorithm works* “ URL:http://neuralnetworksanddeeplearning.com/chap2.html

[76] SydneyF, “*It's a No Brainer: An Introduction to Neural Networks*”, 9 October, (2018), URL:“https://community.alteryx.com/t5/Data-Science-Blog/It-s-a-No-Brainer-An-Introduction-to-Neural-Networks/ba-p/300479”.

[77] Web page “*Logistic Regression”* , URL:https://www.saedsayad.com/logistic_regression.htm

[78] Jason Brownlee “*Logistic Regression For Machine Learning*” , URL : https://machinelearningmastery.com/logistic-regression-for-machine-learning/

[79] Rohith Gandhi " *Support Vector Machine – Introduction to Machine Learning Algorithms* " , URL : https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

[80] Unicode Emoji Charts, "*Full Emoji List,* v13.0" URL:"https://unicode.org/emoji/charts/full-emoji-list.html".

[81] *Emojipedia -Home of Emoji meanings*, emojipedia, Web page URL:"https://emojipedia.org/".

[82] Sriram Chellappan and Raghavendra Kotikalapudi, "*How Depressives Surf the Web*" , URL:https://www.nytimes.com/2012/06/17/opinion/sunday/how-depressed-people-use-the-internet.html

[83] Guntuku, S.C., Schneider, R., Pelullo, A., Young, J., Wong, V., Ungar, L., Polsky, D., Volpp, K.G. and Merchant, R., 2019. *Studying expressions of loneliness in individuals using twitter: an observational study*. BMJ open, 9(11).

[84] Selva Prabhakaran, "*Gensim Tutorial – A Complete Beginners Guide",* October 16, (2018), URL:"https://www.machinelearningplus.com/nlp/gensim-tutorial/".

[85] Scott, H., & Woods, H. C. (2019). *Understanding links between social media use, sleep and mental health: recent progress and current challenges*. Current Sleep Medicine Reports, 5(3), 141-149.

[86] Devitt, A., & Ahmad, K. (2007, June). *Sentiment polarity identification in financial news: A cohesion-based approach*. In Proceedings of the 45th annual meeting of the association of computational linguistics (pp. 984-991).

[87] PyPI, web page, "*gensim 3.8.1",* September 26 (2019), URL:"https://pypi.org/project/gensim/".

[88] Web article "*Beginners Guide To Topic Modeling in Python*" , URL : https://www.analyticsvidhya.com/blog/2016/08/beginners-guide-to-topic-modeling-in-python/

[89] Kumar, A., Sharma, A., & Arora, A. (2019). *Anxious Depression Prediction in Real-time Social Data*. Available at SSRN 3383359.

[90] TextBlob "*API Reference*", URL : https://textblob.readthedocs.io/en/dev/api_reference.html

[91] Web Article, "*These common words and phrases may signal depression"* , URL:"https://www.inc.com/minda-zetlin/these-common-words-phrases-may-signal-depression.html".

[92] Mood Disorders Society of Canada, "*DepressionHurts.ca*", URL:"http://depressionhurts.ca/en/about/".

[93] Shen, J. H., & Rudzicz, F. (2017, August). *Detecting anxiety through reddit.* In Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology—From Linguistic Signal to Clinical Reality (pp. 58-65)

[94] Web article, "*About Account Restoration*", URL : https://help.twitter.com/en/managing-your-account/account-restoration

[95] Tossell, C. C., Kortum, P., Shepard, C., Barg-Walkow, L. H., Rahmati, A., & Zhong, L. (2012). *A longitudinal study of emoticon use in text messaging from smartphones.* Computers in Human Behavior, *28*(2), 659-663.

[96] Web article " Understanding AUC – ROC Curve" , URL: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

[97] Web article *Bell Let's Talk*, URL: "https://letstalk.bell.ca/en/".

[98] Web article "Understanding Confusion Matrix" URL : https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

[99] Wang X., Zhang C., Ji Y., Sun L., Wu L., Bao Z. (2013) *A Depression Detection Model Based on Sentiment Analysis in Micro-blog Social Network*. In: Li J. et al. (eds) Trends and Applications in Knowledge Discovery and Data Mining. PAKDD 2013. Lecture Notes in Computer Science, vol 7867. Springer, Berlin, Heidelberg

[100] Kanakaraj, M., & Guddeti, R. M. R. (2015, March). *NLP based sentiment analysis on Twitter data using ensemble classifiers*. In *2015 3rd International Conference on Signal Processing, Communication and Networking (ICSCN)* (pp. 1-5). IEEE.

[101] Sheoran, Pummy. "*Effectiveness of NLP in Dealing with Guilt Induced Anxiety, Depression and Stress: A Case Study*." *Mental Health: A Journey from illness to wellness* (2016): 179.

[102] Web Article, "*How to Develop Word Embeddings in Python with Gensim*" URL: https://machinelearningmastery.com/develop-word-embeddings-python-gensim/.

[103] Owen Kelly, "The Link Between OCD and Major Depressive Disorder", Januray 10, (2020), URL: "https://www.verywellmind.com/ocd-and-depression-2510591".

[104] Basic evaluation measures from the confusion matrix, URL: “https://classeval.wordpress.com/introduction/basic-evaluation-measures/”

[105] Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PloS one, 10(3).

[106] Mandrekar, J. N. "Receiver operating characteristic curve in diagnostic test assessment." Journal of Thoracic Oncology 5, no. 9 (2010): 1315-1316. URL: “https://www.sciencedirect.com/science/article/pii/S1556086415306043”.

[107] American Psychiatric Association. (2013). Diagnostic and statistical manual of mental disorders (5th ed.),“https://doi.org/10.1176/appi.books.9780890425596”.

[108] Statistics Canada, URL: https://www.statcan.gc.ca/eng/start

Appendix

Hardware Requirements for This Research

Hardware Overview
1. Model Name: MacBook Pro
2. Model Identifier: MacBookPro16,1
3. Processor Name:8-Core Intel Core i9
4. Processor Speed: 2.4GHz Boosted up to 2.5GHz
5. Number of Processors:1
6. Total Number of Cores:8
7. L2 Cache (per Core):256 KB
8. L3 Cache:16 MB
9. Hyper-Threading Technology: Enabled
10. Total Memory: 64 GB (RAM)
11. Activation Lock Status: Disabled

Chipset Model: AMD Radeon Pro 5500M
1. Type: GPU
2. Bus: PCIe
3. PCIe Lane Width:x16
4. VRAM (Total):8 GB
5. Vendor: AMD (0x1002)
6. EFI Driver Version:01.01.190
7. Automatic Graphics Switching: Supported
8. GMux: Version:5.0.0
9. Metal: Supported, feature set mac OS GPUFamily2 v1

Memory Slots Overview
1. ECC: Disabled
2. BANK 0/ChannelA-DIMM0:
3. Size:32 GB
4. Type: DDR4
5. Speed:2667 MHz
6. BANK 2/ChannelB-DIMM0:
7. Size:32 GB
8. Type: DDR4
9. Speed:2667 MHz