Republic of Iraq
Ministry of Higher Education and Scientific
Research University of Babylon
College of Information Technology
Software Department

Predicting Depression in Users on Social Media Using Machine Learning Techniques

A Thesis

Submitted to the Council of the College of Information Technology for Postgraduate Studies of the University of Babylon in Partial Fulfillment of the Requirements for the Degree of Master in Information Technology – Software

By

Rula Kamil Hassan Abbas

Supervised by

Prof.Dr. Ayad Rodhan Abbas

2023 A.D. 1444 A.H.

﴿ٱلرَّحۡمَٰنُ ۝ عَلَّمَ ٱلۡقُرۡءَانَ ۝ خَلَقَ ٱلۡإِنسَٰنَ ۝ عَلَّمَهُ ٱلۡبَيَانَ ۝ ٱلشَّمۡسُ وَٱلۡقَمَرُ بِحُسۡبَانٖ﴾

Declaration

I hereby declare that this thesis entitled " **Predicting Depression in Users on Social Media Using Machine Learning Techniques** ", submitted to the University of Babylon in partial fulfillment of requirements for the degree of Master in Information Technology \ Software, has not been submitted as an exercise for a similar degree at any other University. I also certify that this work described here is entirely my own except for experts and summaries whose source is appropriately cited in the references.

Signature:

Name: **Rula Kamil Hassan**

Date: / / 2023

Supervisor Certification

I certify that the thesis entitled "**Predicting Depression in Users on Social Media Using Machine Learning Techniques**" was prepared under my supervision at the department of Software / College of Information Technology / the University of Babylon as partial fulfillment of the requirements of the degree of Master in Information Technology - Software.

Signature:

Supervisor Name: **Prof. Dr. Ayad Rodhan Abbas**

Date: / / 2023

The Head of the Department Certification

In view of the available recommendations, I forward the thesis entitled "**Predicting Depression in Users on Social Media Using Machine Learning Techniques**" for debate by the examination committee.

Signature:

Prof. Dr. Ahmed Saleem Abbas

Head of Software Department

Date: / / 2023

Certification of the Examination Committee

We, the undersigned, certify that (**Rula Kamil Hassan**) candidate for the degree of Master in Information Technology - Software, has presented his thesis of the following title (**Predicting Depression in Users on Social Media Using Machine Learning Techniques**) as it appears on the title page and front cover of the thesis that the said thesis is acceptable in form and content and displays a satisfactory knowledge of the field of study as demonstrated by the candidate through an oral examination held on:

Signature:
Name: Dr. Khalid Ali Hussein
Title: Professor
Date: / / 2023
(**Chairman**)

Signature:
Name: Dr. Ali Hadi Hasan
Title: Assistant Professor
Date: / / 2023
(**Member**)

Signature:
Name: Dr. Wadhah Razooqi Abbood
Title: Lecturer
Date: / / 2023
(**Member**)

Signature:
Name: Dr. Ayad Rodhan Abbas
Title: Professor
Date: / / 2023
(**Member**) // Main supervisor

Signature:
Name: Dr. Hussein A. Lafta
Title: Professor
Date: / / 2023
(**Dean of Collage of Information Technology**)

Dedications

For the one who spent his life for us to be here

..My father..

To the light that illuminates the darkness of my life

..My mother..

To those who stood to help and support me in anytime and anywhere

..My brothers and sisters..

Acknowledgments

Praise be to **Allah** for his unlimited blessings that continue to flow into my life, and because of You, I made this through.

To my supervisor, Dr. **Ayad Rodhan Abbas**: many thanks for all your advice and comments during our research career. I am grateful for your help to make this research in the best possible form and make me an academic who deserves this certificate.

To the inhalation and exhalation which I cannot live without them .. **my parents**... Thank you for every minute of your life that you sacrificed for me.

For my support in this life, my big brother **Hassanein**, who did not leave me in the most difficult circumstances. Thanks for everything.

Unlimited thanks to my **brothers and sisters** who did not stop supporting and helping me throughout this period.

Last but not least, I would like to thank all the kind, helpful and lovely people who helped me directly or indirectly to complete this work and apologize to them for not being able to mention them by name here, but they are in my heart.

Rula Kamil Hassan

Abstract

Nowadays, depression is a common mental illness. Failure to recognize depression early or guarantee that a depressed individual receives prompt counseling can lead to serious issues. Social media allow us to monitor people's thoughts, daily activities, and emotions, including persons with mental illnesses. Currently, machine learning techniques and deep learning are widely used in sentiment analysis. Therefore, it is easy to utilize them for the early detection of depression.

This thesis focused on three main objectives. The first is developing a model for the early detection of depression among Twitter users. The second is studying the impact of some pre-processing steps on the performance of classifiers that were ignored in previous studies. The third is comparing the performance of different machine learning techniques and finding the best technique.

To achieve these objectives, three types of machine learning techniques were performed in this thesis. The first type includes five traditional machine learning which are support vector machine (SVM), light gradient boosting machine (LGBM), and extreme gradient boosting (XGBOOST), logistic regression (LR), and decision tree (DT). The second type contains two deep learning techniques bidirectional long short-term memory (Bi-LSTM), and convolutional neural network (CNN). The third type is hybrid model which combine between one of traditional machine techniques and one of the deep learning techniques. Two dataset were used from the Kaggle website to predict depression.

The experiments on the first dataset showed that first: Bi-LSTM-XGBOOST is better than single models and achieve the highest performance, with 94% for all evaluation metrics. The proposed model can improve the

performance of machine learning techniques and increase the detection rate of depression. Second, applying stemming, replacing (slang words and emoji), and not removing some stop words can enhance the accuracy of traditional machine learning techniques.

While the second dataset presents that LGBM outperformed other techniques with 99% for all evaluation metrics.

The larger size of the second dataset had an important role in obtaining higher results than the first dataset.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Meaning
AdaBOOST	Adaptive Boosting
ANN	Artificial Neural Network
BERT	Bidirectional Encoder Representations from Transformers
Bi-LSTM	Bidirectional Long Short-Term Memory
BoW	Bag of Words
CatBOOST	Category Boosting
CBOW	Continuous Bag Of Words
CNN	Convolutional Neural Network
CSV	Comma Separated Values
DBN	Deep Belief Network
DT	Decision Tree
EFB	Exclusive Feature Bundling
ELMo	Embedding from Language Models
EXGBOOST	Extreme Gradient Boosting
FN	False Negative
FP	False Positive
GBM	Gradient Boosting Machines
GOSS	Gradient-based One-Side Sampling
IDF	Inverse Document Frequency
LDA	Linear Discriminant Analysis
LGBM	Light Gradient Boosting Machine
LR	Logistic Regression
LSTM	Long Short-Term Memory
MDD	Major Depressive Disease
NB	Naive Bayes
NLP	Natural Language Processing
PCA	Principal Component Analysis
RF	Random Forest
RNN	Recurrent Neural Network
SNS	Social Networks Sites
SVM	Support Vector Machine
TF	Term Frequency
TF-IDF	Term Frequency Inverse Document Frequency
TN	True Negative
TP	True Positive
WHO	World Health Organization

List of Algorithms

Chapter One

Introduction

CHAPTER ONE

INTRODUCTION

1.1 Background

Nowadays, depression is one of the most common mental illnesses worldwide and there are 300 million people are affected. During a depressive episode, the patient loses interest or pleasure in daily activity and suffers from tiredness, lack of sleep, eating disorders, hopelessness, and even suicidal ideation [1].

Approximately eight million people die every year due to mental disorders [2] which is the same number of patients who die every year due to cancers [3].

In the worst cases, depression can cause suicide; more than 700,000 people die every year due to suicide. Despite health care and treatment availability, 75% of people do not receive health care in poor and middle-income countries [4].

The psychiatrist can assess the mental health condition of people from their language because language reveals what people think and feel. Therefore, depressed users of social media can be detected by assessing what they write especially, the negative emotional words [5]. 'Depression language' may have a strong effect on its writers, suicidal poets like Sylvia Plath and Kurt Cobain used suicidal words associated with self in their poetries, and then they killed themselves [6].

Depression can be diagnosed by face-to-face medical interview but 70% of the patients would not consult doctors [7]; or it is diagnosed by 'patient health questionnaire', but also the patient may not answer correctly [8,9]; the third way by asking friends and relatives of the patient. These methods are not accurate [10].

Billions of people around the world are users of Facebook, Instagram, and other types of social media. These became an important part of people's life. Most people share their feelings and daily activity with others. Many studies in cognitive science found that extreme behavior like suicide can be predicted by analyzing an individual's writing style [11]. For example, a 16-year-old Malaysian teen died after asking her followers on her Instagram account to choose life or death and most people voted for her to die [12].

Many research shows the important role of social media in monitoring posts and predicting early cases of depression. Online data are recognized as a source of data applicable to healthcare [13]. Twitter is widely employed in the field of sentiment analysis nowadays, despite the diversity of social media. This is because it makes it possible to retrieve several tweets about a particular subject using specified keywords [14].

In addition, traditional machine learning, deep learning techniques and natural language processing (NLP) are widely used to detect depression and sentiment analysis through social media [15]. Machine learning used in medical field to improve the accuracy, precision, and analysis of diagnostics while reducing laborious jobs. Machine learning has the capability to detect mental distress like depression[16].

In recent years, deep learning techniques have been wildly used in the field of depression detection. The benefit of deep learning is the ability to extract features throughout the learning process from huge datasets [17]. Many studies showed that the results of applied deep learning were better than machine learning [18].

Thus, this thesis attempts to build a new model to predict if a user is depressed or not by using a dataset from Twitter and different machine learning and deep learning techniques.

1.2 Related Works

Currently, most previous studies use different social media datasets from Twitter, Instagram, Reddit, and Facebook to predict depression. The datasets included various types of data such as posts, comments, chats, images, and information users such as number of followers, number of followings, and other personal information.

Previous works had shown that automated analysis textual could be used to detect depression by extracting depressive words or sentences in posts or conversations [19,20,21].

Machine learning and deep learning techniques have played a significant role in classifying diseases. Today, many studies dealt with depression on social media and applied various machine learning techniques.

Thus, this section will highlight some important previous studies. The first sub section presents the studies that applied traditional machine learning. The second explains the researches that used deep learning and traditional machine learning. The last subsection shows the studies that preformed hybrid deep learning techniques. Table (1.1) summarizes the previous studies.

1.2.1 Traditional Machine Learning Techniques

Azam et al. [22] tried to create a machine learning model that can analyze language patterns in tweets, and identify users who are having signs of depression. Two traditional machine learning techniques were used. A dataset was collected by scraping tweets directly for persons who suffer from depression through the use of keyword searches. Two traditional machine learning were applied, which are SVM and random forest (RF). The best accuracy obtained was 77% with RF. The study found that the models need to be more accurate and suggested using more features such as age, number of followers, and gender.

Alsagri and Ykhlef [23] identified the depression cases of Twitter users by analyzing their network activities and tweets through an employed machine learning approach. The dataset was collected from 111 user profiles and more than 300,000 tweets. SVM, naive bayes (NB), and DT were performed as classifiers. 10-fold cross-validation was employed to avoid overfitting. The results of the experiments proved that term frequency inverse document frequency (TF-IDF) and first-person pronouns play an important role in determining depression. The best accuracy was 82% with SVM.

Kumar et al. [24] suggested a model to predicate anxious depression through tweets in real-time. The dataset was scrapped using the Twitter API. The classifiers are multinomial naïve bayes, RF, gradient boosting, and ensemble vote. The authors compared the three individual classifier's accuracy against ensemble vote accuracy. The results show that the best accuracy was 85.09% with ensemble vote classifier.

1.2.2 Deep Learning Techniques

This section presents the studies which applied deep learning and then compare them with traditional machine learning.

Amanat et al. [25] tried to develop an early depression detection system, taking advantage of advances in machine learning and data availability for depression. The same dataset of [26] are used in this thesis, was employed in [25]. SVM, NB, and DT were applied as machine classifiers, and recurrent neural network (RNN) – long short-term memory (LSTM), CNN, and deep belief network (DBN) as deep learning. The results of the study show that RNN-LSTM with one hot and principal component analysis (PCA) and extracted important features from tweet lead to the best accuracy with 99%. The authors suggested the applying of the proposed model to a large dataset.

Uddin et al. [27] proposed a model to recognize symptoms of depression through text using traditional machine learning and deep learning. A big dataset was collected from a public Norwegian online youth information channel. It includes two imbalanced datasets. SVM, LR, and DT were used as machine learning classifier and hybrid RNN-LSTM, CNN, DBN, artificial neural network (ANN). The best accuracy was 99% with RNN-LSTM.

Shetty et al. [12] used machine learning techniques to determine if tweets contain any sign of depression. The dataset from the Kaggle website was used. It includes depressed tweets collected from Twitter. The classifiers employed are LR, RF, Bernoulli naïve Bayes, Multinomial naïve Bayes, Linear SVC, Gradient Boosting Classifier, LSTM, and CNN. The results showed that CNN is the best with an accuracy of 95%. The study suggested using more data to improve accuracy.

1.2.3 Hybrid Deep Learning Models

Tejaswini et al. [28] suggested a hybrid model that combines CNN with LSTM to predicate depression in social media. In addition, the fast text embedding method was used to represent text. Two small datasets were used, the first from Reddit and the second from Twitter [26]. The finding for this paper shows that the hybrid model with fast text gets results better than other techniques with an accuracy of 87% with first dataset and 88% with second dataset.

Kour and Gupta [29] suggested a hybrid model (CNN-Bi-LSTM) from two deep learning techniques CNN and Bi-LSTM to predicate depression. The paper used a dataset from Twitter. Also, data preprocessing and word embedding was applied. The proposed model was employed to data, then compared with CNN and RNN techniques. The results show that the suggested model is better than other techniques and achieved an accuracy of 94%. The authors proposed using other combinations from neural network layers and activation functions. The second suggestion is to use pre-trained language techniques such as embedding from language models (ELMo) and bidirectional encoder representations from transformers (BERT) .

Bhargava et al. [18] proposed a hybrid model CNN-LSTM from two deep learning techniques CNN and LSTM to detect depression. The paper used a dataset from Twitter. Preprocessing and word2vec, sentiment extractions by Vader were applied. CNN_LSTM model was applied and compared with LSTM model. The authors found that CNN_LSTM is better than LSTM with an accuracy of 91%.

Mumu et al. [30] proposed a hybrid CNN-LSTM model to detect depression. Authors applied the model to a small dataset of Bengali Facebook. As well, TF-IDF and count vectorizer was employed. Four types of classifiers were performed. Those were SVM, NB, LR, and RF, and CNN-LSTM. Then a comparison between

them. The results prove that CNN-LSTM is better than other algorithm and lead to an accuracy of 81.05.

Dang et al. [17] suggested a hybrid model from a combination of CNN, LSTM, and SVM for sentiment analysis. The author used 8 datasets to test and evaluated the model. Preprocessing and word2vec, BERT as features vector was performed. The results showed that the hybrid model improved the accuracy of sentiment analysis compared with the single model. In addition, the combination of techniques makes able to benefit from the advantages of each technique, where CNN is able to extract features, LSTM can able to keep past information at cell state, and SVM is a good classifier. Using SVM as a classifier to improve the performance of CNN_LSTM and LSTM_CNN models.

Table (1.1) A Summary of the Related Works

Study	Feature extraction and classifier	year	Dataset type	Findings
Kumar et al. [24]	NB, RF, gradient boosting, and ensemble vote,	2019	Twitter	The results show that the best accuracy was 85.09% with ensemble vote classifier. Ensemble vote classifier is better than other individual classifier.
	Features extraction	Word, timing, frequency, sentiment, contrast		
	Replacing slang & emoji	stemming		Customized removing stop words
	yes	yes		No
Study	classifier	year	Dataset type	Findings
Alsagri and Ykhlef [23]	SVM, DT, NB	2020	Twitter	The best accuracy obtained was 82% with SVM. It found that TF-IDF and first-person pronouns play an important role in determining depression.
	Features extraction	TF-IDF and others features		
	Replacing slang & emoji	stemming		Customized removing stop words
	No	yes		First person pronoun only
Study	Feature extraction and classifier	year	Dataset type	Finding
Shetty et al. [12]	LR, RF, Bernoulli naïve Bayes, Multinomial naïve Bayes, Linear SVC, Gradient Boosting Classifier, LSTM, and CNN	2020	Twitter	The best accuracy was 95% with CNN. The study suggested using more data to improve accuracy.
	Features extraction	Count vector, TF-IDF.		
	Replacing slang & emoji	Stemming & lemmatization		Customized removing stop words
	No	yes		No

Study	classifier	year	Dataset type	Findings
Azam et al. [22]	SVM RF	2021	Twitter	The best accuracy obtained was 77% with RF. The models need to be more accurate and suggested using more features.
	Features extraction	bag of words (BOW)		
	Replacing slang & emoji	lemmatization		Customized removing stop words
	Emoji only	yes		First person pronoun only

Study	Feature extraction and classifier	year	Dataset type	Finding
Uddin et al. [27]	SVM, LR, DT, RNN-LSTM, CNN, DBN, and ANN.	2021	public Norwegian online youth information channel	The best accuracy was 99% with RNN-LSTM and one-hot and proposed features.
	Features extraction	One hot, TF-IDF, Linear discriminant analysis (LDA) and proposed important features extracted from dataset.		
	Replacing slang & emoji	stemming		Customized removing stop words
	No	yes		No

Study	Feature extraction and classifier	year	Dataset type	Finding
Bhargava et al. [18]	CNN-LSTM, LSTM	2021	Twitter	CNN_LSTM model was applied and compared with LSTM model. The authors found that CNN_LSTM is better than LSTM with an accuracy of 91%.
	Features extraction	word2vec, sentiment extractions by Vader.		
	Replacing slang & emoji	lemmatization		Customized removing stop words
	Slang words only	yes		No

Study	Feature extraction and classifier	year	Dataset type	Finding
Mumu et al. [30]	NB, SVM, LR, RF, and CNN-LSTM.	2021	dataset of Bengali Facebook	The results prove that CNN-LSTM is better than other algorithm and lead to an accuracy of 81.05.
	Features extraction	Count vectorizer, TF-IDF.		
	Replacing slang & emoji	Stemming		Customized removing stop words
	No	yes		No

Study	Feature extraction and classifier	year	Dataset type	Finding
Dang et al .[17]	CNN, LSTM, and SVM.	2021	8 datasets from tweets and review datasets of different domains.	The results showed that the hybrid model improved the accuracy of sentiment analysis compared with the single model.
	Features extraction	Word2vec, BERT.		
	Replacing slang & emoji	Stemming & lemmatization		Customized removing stop words
	No	No		No

Study	Feature extraction and classifier	year	Dataset type	Finding
Amanat et al .[25]	SVM, NB, and DT, CNN, DBN, RNN-LSTM	2022	Twitter	The best accuracy was 99% with RNN-LSTM and one-hot and proposed features. The authors suggested applied hybrid neural networks on large dataset.
	Features extraction	One hot, TF-IDF, PCA and proposed important features extracted from dataset.		
	Replacing slang & emoji	Stemming & lemmatization		Customized removing stop words
	No	yes		No

Study	Feature extraction and classifier	year	Dataset type	Finding
Tejas wini et al. [28]	CNN-LSTM, CNN and LSTM-RNN	2022	Twitter, Reddit	The finding for this paper shows that the hybrid model CNN-LSTM with fast text gets results better than other techniques with an accuracy of 87% with first dataset and 88% with second dataset.
	Features extraction	Fast text, Glove, word2vec		
	Replacing slang & emoji	Stemming & lemmatization		Customized removing stop words
	No	yes		No

Study	Feature extraction and classifier	year	Dataset type	Finding
Kour and Gupta [29]	CNN-Bi-LSTM, CNN, RNN.	2022	Twitter	The results show that CNN-Bi-LSTM is better than other techniques and achieved an accuracy of 94%.
	Features extraction	Word embedding.		
	Replacing slang & emoji	stemming		Customized removing stop words
	No	yes		No

1.3 Research Problem

Many studies proposed to detect depression. However, There are some limitations in the existing works:

1. Ignoring some important pre-processing steps, such as eliminating essential stop words typically used by depressed people, negation words and pronouns, slang words replaced with comparable words, and emoji replaced with text. This pre-processing can increase accuracy of predictions..
2. Some previous studies obtained high accuracy but used imbalanced labeling datasets or small datasets.
3. Most of the previous studies use one dataset, and this may generate a weak system.

1.4 Thesis Questions

This thesis tries to answer the following questions:

1) Is the pre-processing steps can impact and enhance the performance of classifiers? what are these pre-processings?

2) Which is better, traditional machine learning or deep learning techniques for predicting depression in social media?

3) Is the performance of the proposed system will be enhanced by the combination of deep learning and traditional machine learning techniques?

4) Is prediction accuracy affected by the number of tweets?

1.5 Aims of Thesis

The main objective for this thesis is creating an efficient model for detecting depression among users of social media.

1) Obtaining a higher accuracy than previous studies.
2) Finding the most effective pre-processing techniques to improve performance.
3) Finding the best technique among machine learning techniques.
4) Applying the system to more than one dataset to test the efficiency of the proposed system.

1.6 Thesis Significance and Contributions

1) Computing execution time for machine learning techniques. This ignored in previous studies.

2) Some preprocessing steps are suggested to enhance the accuracy of traditional machine learning techniques such as customized removing stop words, and replacing slang words & emoji that were ignored in previous studies.

3) New hybrid models were suggested that mix the architecture of the machine with deep learning techniques and it has produced good results. This has not been suggested in most previous research.

1.7 Thesis Challenges

During various stages of this thesis, there were different challenges:

- Difficult obtaining a dataset that contains real depressed tweets, or the dataset is not appropriate because is small or unbalanced, where the number of non-depressed tweets exceeds the number of depressed tweets.

1.8 Thesis Organization

This thesis is organized as follows:

- Chapter Two (Theoretical Background): This Chapter presents the overview about depression, social media, preprocessing steps.
- Chapter Three (Proposed System) : The proposed system and its algorithms are covered.
- Chapter Four (Results and Discussion): clarifies the outcomes of the thesis experiments and the suggested system. Additionally, the performance evaluation of the system is covered.
- Chapter Five (Conclusions and Future Works): The thesis results are summarized in this chapter, along with some possible future study directions.

Chapter Two

Theoretical Background

CHAPTER TWO

THEORETICAL BACKGROUND

2.1 Introduction

This chapter includes the theoretical background of depression, traditional methods of diagnosis, and the disadvantages of these methods. In addition, it presents the possible solutions that the computer field can offer to diagnose depression, such as using of social media and machine learning techniques.

After that, explains the concepts of machine learning techniques that are applied, and NLP. In addition, it reviews the most important steps in the processing of tweets such as preprocessing, features extraction, the metrics performance for classification algorithms.

2.2 Depression

Depression is a popular mental disorder [31]. Many types of depressive disorder, and for each type, there are a set of symptoms. Major depressive disease (MDD) is a common type of depressive disorder. With this type, people cannot work, eat, or sleep. The patient must suffer from five or more of the set of symptoms for at least two weeks and every day. The symptoms includes a sad mood for the most of day, lack of interest in all activities, losing or gaining weight, lack of energy, body agitation or retardation, feeling of remorse or worthlessness, cannot sleep or sleep for more time, thought of death and suicide [32].

Nowadays, depression is a widespread illness. There are many people worldwide who suffer from depression. According to the world health organization (WHO), there are more than 264 million people of all ages suffer from depression [33]. Depression is one of the most causes of suicide, with over 800.000 suicide

deaths occurring every year, also is the second cause of death among individuals 15-29 years old range [31].

Therefore, early diagnosis plays an important role in reducing the severity of the disease. Many methods are used in the medical field to detect depression such as interviews, questionnaires, and asking the patient's family or his friends.

However, early detection of depression by traditional methods can be difficult for several reasons. First, depression can be diagnosed by face-to-face medical interview but 70% of the patients would not consult doctors [7]; or it is diagnosed by 'patient health questionnaire', but also the patient may not answer correctly [8,9]; the third way by asking friends and relatives of the patient. These methods are not accurate [10].

Another reason is, sometimes the traditional methods of treatment, such as drugs and psychotherapy, are expensive and ineffective, therefore the patient does not want to consult a doctor. Second, the patient's fear of stigma after being diagnosed with depression [34].

Thus, many studies suggested using other approaches. One of the most methods is social media. Social media is widely used by many people. Therefore, this thesis resorted to using social media.

2.3 Social Networks Sites (SNS)

Social networks sites can be defined as web-based services that enable people to create public or semi-public profile, list of other people with whom share connection within the system, view and traverse their list of connection [35]. Now, SNS became widely used, many people around the world use it in daily life, such as Facebook, Twitter, and LinkedIn [36].

SNS presented many technical features that support a wide range of interests and practices such as the maintenance of preexisting social networks, connections between different people based on shared interests, political views, or activities. Some sites may be for anyone and other sites attract people based on common language or shared racial, sexual, religious, or nationality based identities. Some sites support incorporate new information and communication tools, such as mobile connectivity, blogging, and photo/video-sharing. Among all these sites, Twitter is the fastest growing one than others sites [36]. This thesis uses Twitter as site because the widespread use of it.

2.3.1 Depression in SNS

Many methods used in medical fields to diagnose depression such as surveys, and interviews, but it are not accurate. Despite abroad health care programs and treatment availability, the detection rate is low and most depressed people deny looking for treatment.

In recent years, the usage of social media is common, especially, among younger [32]. In 2015, there were about 2 billion social media users, and it is growing every day [37]. Users can use SNS at any time and from any location through their mobile or computer. The availability of social media enabled users to share feelings, interests, and daily life. The social media became the safe space for people to share negative feelings and emotions [32].

Social media has become a data source in different contexts, especially in the medical field to monitor people's health such as mental health. User-generated content on social media may be used to detect well-known symptoms of depression and this represents a new form for the screening of the mental disorder [33]. Many studies mention that language patterns may be indicators of mental state and use in the early detection of depression [32,33,38]. Thus, previous studies used social

media in the early detection of depression. This thesis, used Twitter to detect depression through text.

2.3.2 Twitter

Twitter is an online microblogging social media platform that enables users to share short messages about events, emotions, and ideas. With more than 330 million active users worldwide, Twitter is one of the most significant social media platforms [39].

Twitter was created by Jack Dorsey, Noah Glass, Biz Stone, and Evan Williams in March 2006, and the full version of the service was made available for free in July of the same year. Twitter was inspired by the desire to use a short messaging system for a small group. The first name for this project is Twttr, but it was soon changed to Twitter. In addition, users of Twitter can view additional media content by clicking on tweets that contain links to videos and images from various websites [40].

2.4 Machine Learning Techniques

It is a subfield of artificial intelligence. It means automated learning, where computers can learn from input data. Learning is converting experiences to knowledge and expertise. The training data represent experiences, while the outputs act as expertise [41]. Generally, there are three main types of ML: supervised, unsupervised and semi-supervised techniques. Supervised machine learning is based on labeled datasets in the training stage to learn themselves; this enables it to predict the input data in future, such as classifying email as spam and not spam. On the other hand, unsupervised techniques do not need to label the dataset; the aim is not to classify data but to discover hidden data and find structure or organize it. Semi-supervised is a combination of supervised and unsupervised techniques where the dataset includes a small amount of labeled data and the others

with no label [42]. Semi-supervised usually used to label datasets without labels because labeling data requires a human expert, and this is expensive. Figure (2.1) illustrates the types of machine learning techniques.

In this thesis, supervised techniques were used to classify the unseen input data. It includes the use of traditional methods such as SVM, DT, and extreme gradient boosting (XGBOOST). In addition, deep learning techniques were used such as LSTM, and CNN.

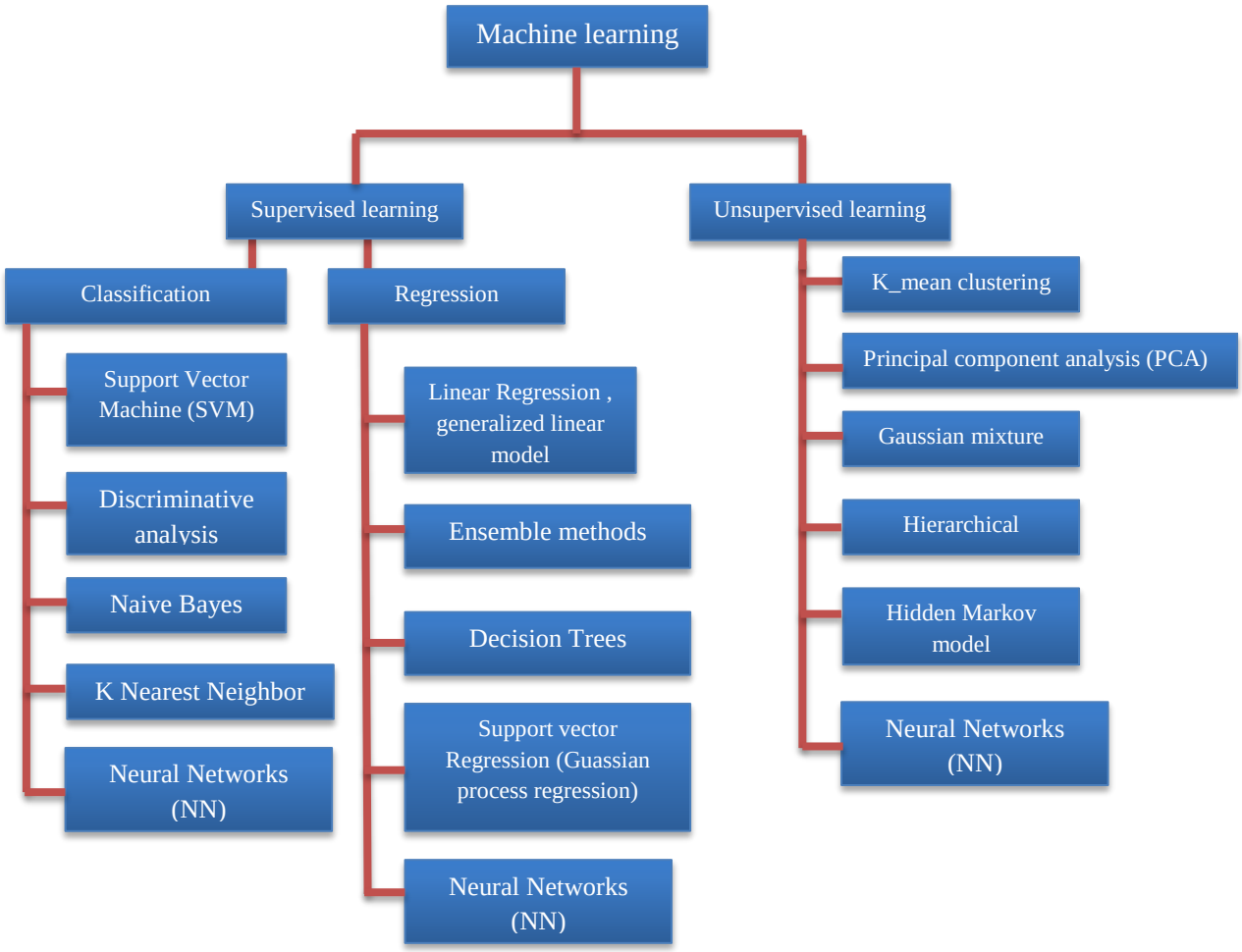


Figure (2.1) Types of Machine Learning Techniques [43]

2.4.1 Traditional Machine Learning Techniques

2.4.1.1 Support Vector Machine (SVM)

SVM is one of the most supervised machine learning techniques. Its widely used to classify text and images categorization or hand-written recognition. The goal of the linear SVM model is to classify objects into two main classes using a classifier [44]. Figure (2.2) shows the work of linear SVM.

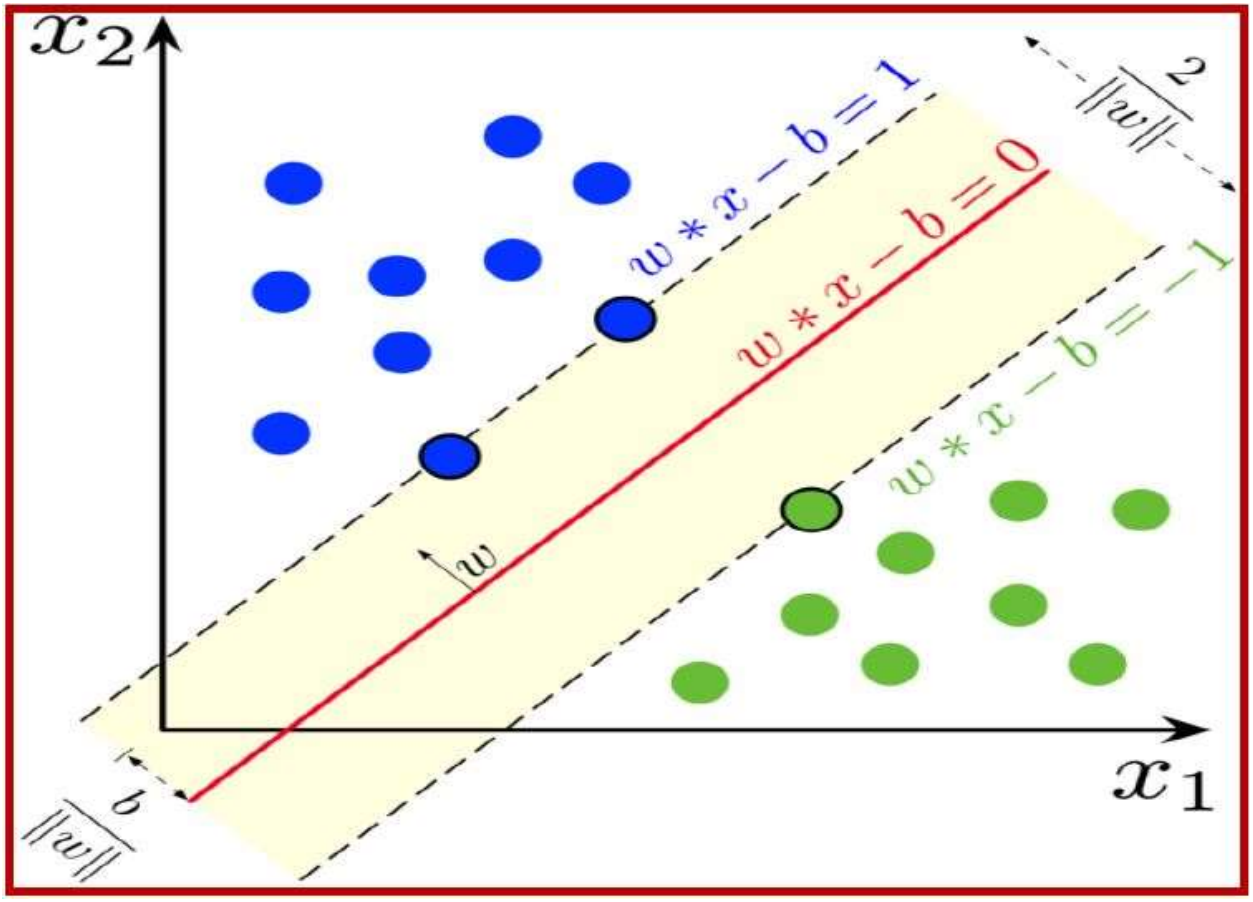


Figure (2.2) SVM Algorithm [40]

To explain how linear SVM classify two class. Consider, there is a binary classification problem has N training samples. Each sample has tuple (x_i, y_i) where x_i $\{x_i,\ldots,x_n\}$refers to features and $y_i \in \{1,-1\}$ refers to the class label [40]. According to Figure (2.2) ,there are a set of mathematical equations.

Equation 2.1 to find the decision boundary.

$$f(x)= W^T.X_i +b \qquad (2.1)$$

Where b is a bias term and w denotes a weight vector used to define the decision boundary. If f(X) =1 then x is belong to the first class else if f(x)= -1 then x belong to second class. f(x) = 0 determines the optimal hyperplane [44].

The margin of the decision boundary is given by the distance between these two hyperplanes . See Equation 2.2 [40].

$$d= 2/||w|| \qquad (2.2)$$

To learn SVM model, the parameter b and w must be select in the method that achieved the following two condition in Equations 2.3 and 2.4.

$$W^T. X_i +b \geq 1 \text{ for } y_i=1 \qquad (2.3)$$

$$W^T. X_i +b\leq 1 \text{ for } y_i= -1 \qquad (2.4)$$

The decision boundary must classify all points correctly. Both inequalities can be summarized in a more compact form by Equation 2.5 [40]:

$$y_i (W^T. X_i +b) \geq 1, i=1,2,..,N \qquad (2.5)$$

Algorithm (2.1) explains SVM pseudocode [45].

The SVM is effective tool to classification text. Therefore, this thesis use it.

Algorithm (2.1): General SVM Algorithm [45]

```
1.Initialize the weights and bias:
    w = zeros(d) # d is the number of features
    b = 0
2. Choose the hyperparameters:
    C: regularization parameter
    learning_rate: step size for gradient descent
    epochs: number of times to loop over the dataset
3. Train the SVM using gradient descent:
        for epoch in range(epochs):
            for i, (x, y) in data:
                if y * (dot product (x, w) - b) >= 1:
                    w -= learning_rate * (2 * C * w)
                else:
                    w -= learning_rate * (2 * C * w – dot product(x, y))
                    b -= learning_rate * y
4. Make predictions:
        for x in test_data:
        prediction = sign(dot product (x, w) - b)
```

2.4.1.2 Gradient Boosting Machines (GBM)

Gradient boosting is one of ensemble learning algorithms. It combines a set of weak learner to build one strong leaner [46]. Due to its accuracy and efficiency it has become widely used. The beginning of it was with adaptive boosting (AdaBOOST) then after, it developed to many techniques such as category boosting (CatBOOST), XGBOOST, and LightGBM [47].

2.4.1.3 Extreme Gradient Boosting (XGBOOST)

XGB is an ensemble tree-based techniques that applied a gradient boosting machine learning framework to solve classification and regression issues. The level-wise methods are used by XGB to grow trees. There are difference between XGB and random forest in grow of tree, orders, and combine the results [46].

Different algorithms are used by XGB to find splits such as exact greedy and approximate algorithms that are presented first then histogram-based algorithm appeared after the LGBT method was developed. The loss value is used to determine if a split is happening or not. If the loss value exceeds a specific threshold value then the split happens else it will ignore. This is one of the advantages leaf-wise in minimizing the number of splits with keeping the quality of a split [46].

2.4.1.4 Light Gradient Boosting Machine (LGBM)

One of most gradient boosting algorithms that based on a decision tree. It was proposed by Microsoft in 2017. The tree grows vertically by using a leaf-wise algorithm as explains in Figure (2.3). Mostly, the leaf that minimizes the loss is selected for tree splitting and growth. The best split candidates are found using the LGBM histogram-based technique [46]. It was characterized by the speed in training, needs for lesser memory, and compatibility with big datasets.

The light in LightGBM gets from the fact that it is fast and uses very little memory. Additionally, it supports GPU learning [48].

LGBM is developed to get more speed, accuracy, and small memory usage better than XGB. The main difference between LGBM and XGB is the growth of the tree. LGBM used a leaf-wise method by splitting leaf nodes using a histogram method that lead to significant advantages in memory usage and efficiency. While XGB is used level-wise. The complexity of the model increases as a leaf-wise tree

grows, therefore LGBM can produce greater accuracy gains with each iteration. However, this can lead to overfitting [49]. Algorithm (2.2) explains generic ligthtGBM.

Algorithm (2.2) explains generic ligthtGBM [50].

Input :

Training data: D={(x1,y1), (x2,y2),…, (xN,yN)}, xi ϵX, x⊆ R, yiϵ{-1,+1}; loss function :L(y,Θ(x));

Iterations: M; Big gradient data sampling ratio: a; slight gradient data sampling ratio :b;

1:Combine features that are mutually exclusive (i.e., features never simultancously accept nonzero values) of xi, i={1,…, N} by the exclusive feature bundling (EFB) technique;

2: Set $\Theta_0(x)$= arg $\min_c \sum_i^N L(yi, c)$;

3: For m=1 to M do

4:Calculate gradient absolute values:

Ri= $\left|\frac{\partial L(yi,\theta(xi))}{\partial \theta(xi)}\right|_{\Theta(x)=\Theta m\text{-}1(x)}$,i={1,…,N}

5: Resample data set using gradient-based one-side sampling (GOSS) process:

topN = a X len(D); randN = b X len(D)

sorted =GetSortedIndices(abs(r));

A=sorted [1: topN]; B=RandomPick(sorted[topN: len(D)], randN);

D'=A+B;

6: Calculate information gains:

$V_i(d)=\frac{1}{n}\left(\frac{\left(\sum xi\epsilon Ai\ ri+\frac{1-a}{b}\sum xi\epsilon Bi\ ri\right)2}{ni1\ (d)}+\frac{(\sum xi\epsilon Ar\ ri+\frac{1-a}{b}\sum xi\epsilon Brri)2}{n1r(d)}\right)$

7: Develop a new decision tree $\Theta_m(x)$' on set D'

8:Update $\Theta_m(x)=\Theta_{m-1}(x)+\Theta_m(x)$

9:End for

10: Return θ'(x)=$\Theta_M(x)$

To make the LGBM a quick, effective, and reliable algorithm, two techniques—gradient-based one-side sampling (GOSS) and exclusive feature bundling (EFB) are also used. Because data samples with high gradients would contribute more to the information gain during training, GOSS chooses the high gradient samples and discards the low gradient ones. However, EFB minimizes

dimensionality and boosts efficiency by bundling unique features in a small amount of feature space [49].

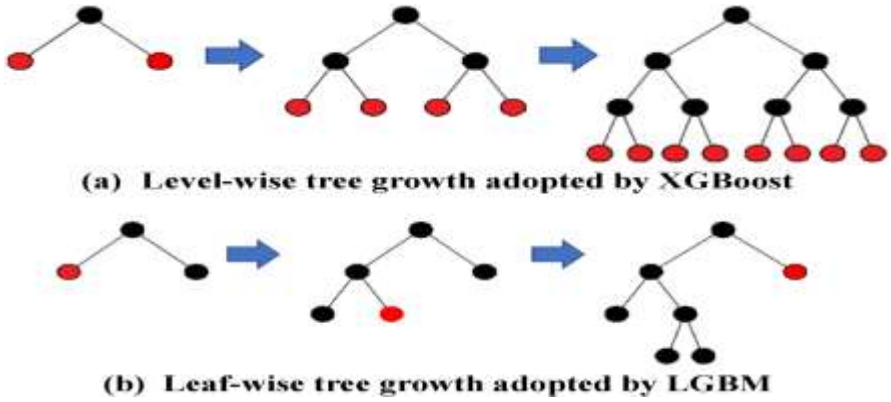


Figure (2.3) Types of Tree Growth [49]

2.4.1.5 Logistic Regression (LR)

One of the supervised machine learning techniques and a statistical method. logistical regression are common regression algorithms [51,52]. The word "regression" in the name is derived from linear regression, a close type in the regression field [53]. LR is more of a classification model than a regression model [52]. For binary and linear classification issues, LR is a simple and more effective solution.

It is employs to classify binary problems with a label (0, 1) or (True, False). It is use the sigmoid function to find the probability of class labels [51,52]. While utilizing the softmax function for multi-class problems [46].

In supervised classification issues, the classes are discrete, therefore the algorithms' objective is to identify the decision boundaries between the classes. Decision boundaries may have a complex and nonlinear geometric shape depending on the specific problem instance. The classes are separated by decision boundaries [53].

The best approach to understand LR is to think of it as linear regression applied to classification issues. The main difference between linear and logistic is that the value for logistic between 0 and 1. Furthermore, with LR there is no need to linear relationship between inputs and output variables [54].

2.4.1.6 Decision Tree (DT)

One of the most popular machine learning techniques. It creates a prediction model as a tree by splitting the dataset into smaller subgroups [40]. Each internal node in the tree represents a test for specific features, each branch represents results for the test, and the leaf node deontes to the label of class. According to the results of the tests along the path, samples are classified by transferring them from the tree's root down to a leaf [54]. Figure (2.4) shows DT.

On its accuracy, the tree complexity plays a critical role. The stopping criteria and pruning technique are specifically utilized to control the tree's complexity. The total number of nodes, the total number of leaves, the tree depth, and the number of features are commonly used to measure the complexity of a tree [54]. The rule for DT is shown in Equation 2.6 [40].

$$Entropy(t) = -\sum_{i=0}^{c-1} p(i|t) log2\mathrm{p}(\mathrm{i}|\mathrm{t}) \quad (2.6)$$

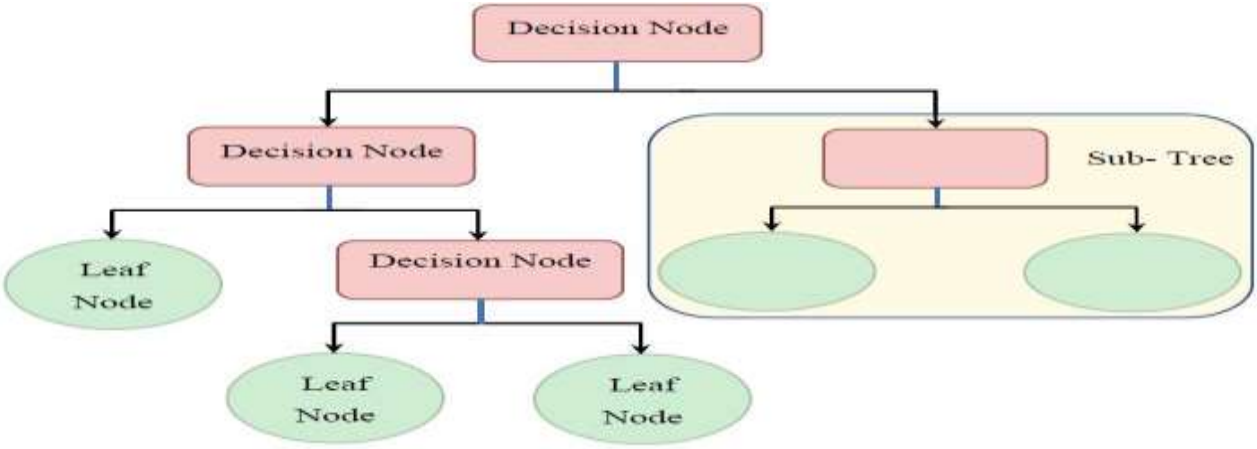


Figure (2.4) Main Idea for DT [40]

Where c is the number of classes, (p|t) denotes the fraction of records belonging to class *i* at a given node t [40].

There are some algorithms that belong to DT such as ID3, C4.5, CHAID, and CART. The most important benefit of this method is its ability to handle both categorical and numerical attributes. For small datasets, this approach works well, but for large datasets, it lags [55].

2.4.2 Deep Learning Techniques

Nowadays, the deep learning widely used in many fields. Deep neural networks usually have multiple hidden layers as in Figure (2.5) that are arranged in layered network architectures. In addition, they often include more complex neurons than simple ANNs. Rather than employing a simple activation function, they may instead involve sophisticated processes (such as convolutions) or several activations in a single neuron [56].

It also known as representation learning, is a set of algorithms that, when given an original dataset, automatically determine the classification or detection that is required. The main drawback of traditional machine learning is the selectivity-invariance problem, which limits their ability to process the input in their original form [55].

It is subfield of machine learning. The main difference between deep learning and conventional pattern recognition is that deep learning rather than dependence on handcrafted features, automatically learns features from big data [57]. Because deep learning can handle a lot of features while working with unstructured data, it has greater power and flexibility. In deep learning techniques the data move from one layer to other. Each layer can able to extract features and

then send to the next layer. Low-level features are extracted by the first layers, which are followed by layers that integrate features to create a full representation.

Deep learning technology is used in many NLP tasks such as sentimental analysis, information retrieval, semantic parsing, question answering, semantic role labeling, machine translation, text generation, relation extraction, summarization, event detection, and text classification [57]. Most widely utilized deep neural networks are CNN and RNN. Therefore, this thesis used CNN and Bi-LSTM.

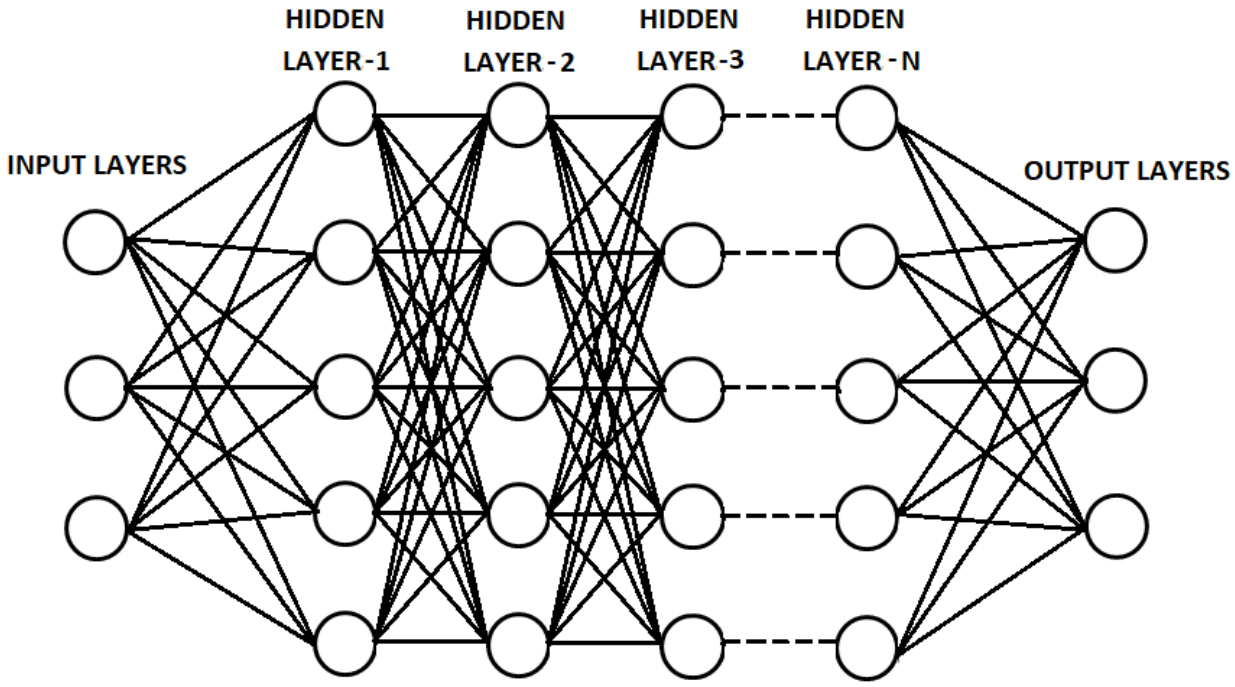


Figure (2.5) Architecture of Deep Learning [55]

2.4.2.1 CNN (Convolutional Neural Network)

CNN is a type of deep feed-forward neural network that employs multilayer perceptrons with minimum preprocessing. CNN was first designed for image classification and computer vision problems. Now, it used in many NLP tasks. In the NLP task, when CNN is applied to text, there is a need to represent the text as a 1D array. The 1D convolutional and pooling process is part of the CNN

architecture. ConvNet uses n-grams to classify a sentence into a number of predefined classes [58].

There are two parts to the CNN layer: feature extraction and classification. In feature extraction, convolution series, and pooling processes are performed. In classification, a fully connected person will complete their work. It applies a filter to the input data to create a feature map. After padding the feature map, the convolutional layer is created. Max pooling is performed as a pooling layer to reduce the size of the feature map by selecting the largest value from each window. The fully connected layer is used to transform data from 2D or 3D to 1D. The fully connected layer is the last layer and is used to show the output of the network [30]. CNN is a well-known deep learning architecture that doesn't require human feature extraction and learns directly from input [59]. Figure (2.6) shows structure of CNN. Thus, this thesis used CNN.

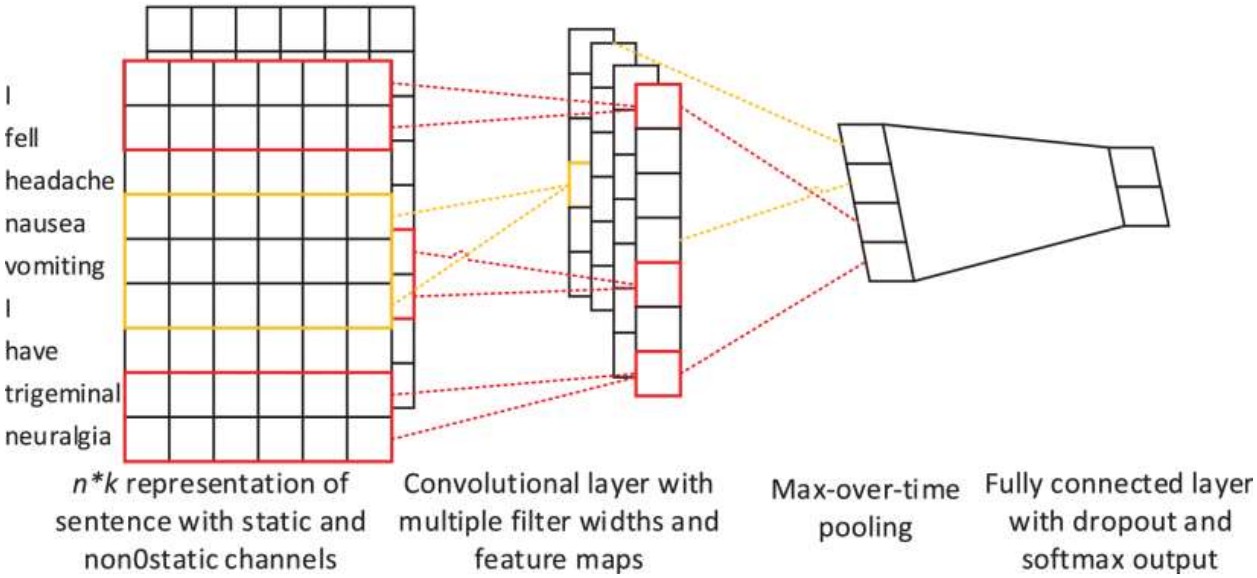


Figure (2.6) Structure of CNN [60].

2.4.2.2 Bi-LSTM (Bidirectional Long Short-Term Memory)

Bi-LSTM is a type of RNN. Before talking about Bi-LSTM, it is necessary to know what RNN is. RNN is a type of neural network. It uses time-series or sequential data such as video or text. In contrast to feedforward neural networks,

the output from the past state is fed to the current state. RNN learn from training input as CNN and feedforward [59].

RNN is characterized by “memory” which allows it to influence current input and output through using information from previous inputs [59]. The memory helps the network to easily distinguish patterns that are used in voice translation, stock predictions, language translation, and image recognition [61].

RNN's architecture consists of input as text that can include a variety of features and hidden layers, as well as an output with respect to time. When using forward propagation, the RNNs preprocess one word at a time and attempt to keep the order of the words by combining the previous words with the new words as a loop and feeding them as input to the hidden layers. However, there is a problem with RNN, vanishing gradients that make it difficult to learn from long data sequences [61]. The cause for that is the memory remembers the previous data but for short time.

Thus, LSTM is developed. LSTM is common type for RNN architecture that use special unit to solve vanishing gradients issue. An LSTM unit's memory cell has the ability to store data for a long time, and three gates control the information's flow in and out of the cell. The "Forget Gate," for instance, decides what data from the previous state cell will be remembered and what data will be deleted that is no longer beneficial. While the "Output Gate" identifies and controls the outputs, the "Input Gate" decides which information should enter the cell state [61].

However, LSTM can keep just the past information. Bi-LSTM was suggested to solve this issue, since Bi-LSTM can keep the context for previous and future related information [62]. In NLP tasks, bidirectional LSTM is a common

option [61]. Therefore, this thesis used Bi-LSTM. Figure (2.7) explains structure of Bi-LSTM.

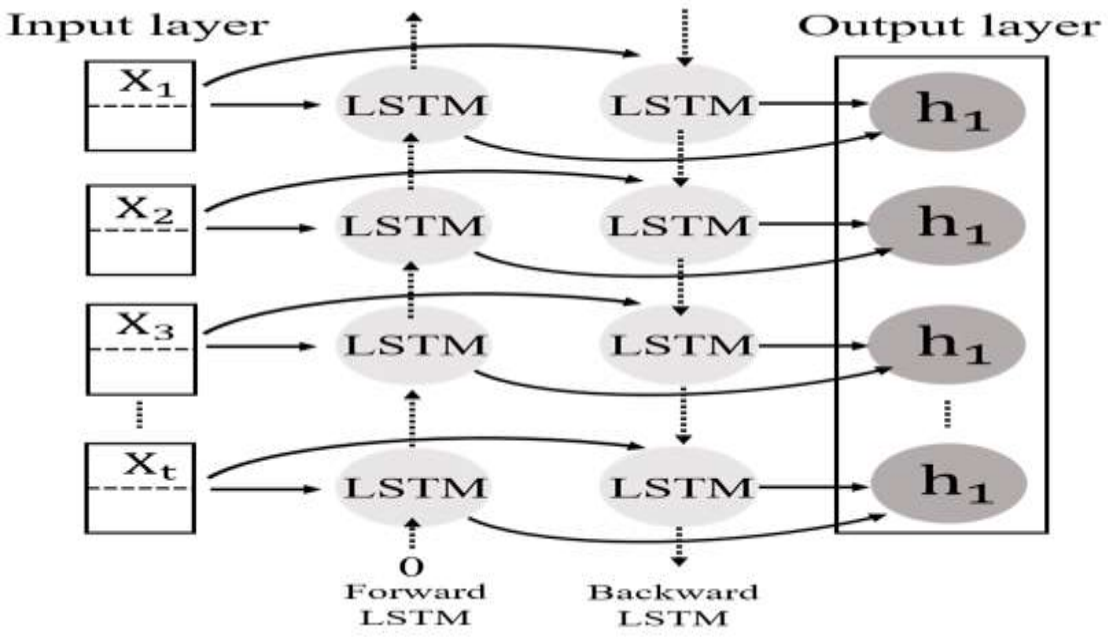


Figure (2.7) the structure of Bi-LSTM [63]

2.5 Natural Language Processing (NLP)

It is a subfield of artificial intelligence. NLP is a field of study and application that searches how computers can understand and process natural language text or speech to be useful things. NLP has applications in a number of fields such as machine translation, speech recognition, artificial intelligence and expert systems, natural language text processing and summarization, and so on [64].

It is difficult for machine learning to understand natural language directly so it must convert to an appropriate form. NLP provides a set of techniques such as text preprocessing and features extraction methods to make the text more understood by machines.

2.5.1 Text Preprocessing

Machine learning simply understands the language of numbers; it is unable to interpret or recognize text or strings as humans do. Finding techniques such as text preprocessing to help the machine understand the text adequately is therefore necessary.

The raw data that reads from the CSV file contains tweets. The tweet may involve unwanted data such as special characters like @, #, * or numbers, links to videos or images, capital letters, or stop words. All these unnecessary data are processed by text preprocessing.

Data pre-processing techniques play an important role in cleaning up unwanted data that are not useful for the machines to interpret [65]. There are many techniques for text preprocessing such as tokenization, stemming, lemmatization, below mention some important of them that used in this thesis.

The tweet passes through a number of process stages. One of important preprocessing is tokenization. Tokenization is a process of splitting sentences into individual words, characters, and punctuation marks called tokens. The major dividing criterion is the presence of a space or a punctuation mark. This procedure aids in removing unnecessary words for later processing phases [65].

Another preprocessing is cleaning tweets from unwanted data as links, numbers, punctuations, retweets, mentions, unrecognized emoji, and symbols must remove. Convert upper case letters to lower case to be the words in the same form. In addition, emoji and emoticons should be converted rather than removed from text because they express feelings like sadness or happiness. Instead, the appropriate words should be used in their place. For instance, the emoticon :-) becomes smiley and :-(becomes sad, and emoji 💕 becomes two hearts. Table (2.1) shows emoticon with corresponding word.

Table (2.1) Emoticon with Corresponding Word [66].

emoticon	Corresponding word	emoticon	Corresponding word
:-)	smiley	:<	sad
:-]	smiley	:[	sad
:-3	smiley	:-‖	sad
:->	smiley	>:[	sad
8-)	smiley	:{	sad
:-}	smiley	:@	sad
:)	smiley	>:(	sad
:]	smiley	:'-(	sad
:3	smiley	:'(	sad
:>	smiley	:-P	playful
8)	smiley	X-P	playful
:}	smiley	x-p	playful
:o)	smiley	:-p	playful
:c)	smiley	:-Þ	playful
:^)	smiley	:-þ	playful
=]	smiley	:-b	playful
=)	smiley	:P	playful
:-))	smiley	XP	playful
:-D	smiley	xp	playful
8-D	smiley	:p	playful
x-D	smiley	:Þ	playful
X-D	smiley	:þ	playful
:D	smiley	:b	playful
8D	smiley	<3	love
xD	smiley		
XD	smiley		
:-(	sad		
:-c	sad		
:-<	sad		
:-[	sad		
:(	sad		
:c	sad		

Most people tend to use slang words or shortcuts when speaking, which may make it difficult for the classifier to distinguish between several terms with the same meaning. So, it must replace with stander words. Table (2.2) display shortcut and slang words with corresponding word.

There are some unimportant words that appear frequently in text such as (is, the, and in). These words, which are known as stop words, are utilized to connect words in a phrase, that really doesn't mean anything [67]. Removing stop words will reduce the size of the text to be processed by the classifier.

Another pre-processing is stemming. This process is a type of linguistic normalization, that tries to reduce the various word forms into a common form. For instance, the term "organize" is a common representation of the terms "organize", “organized”, “organizing”, “organization”, and "organizations" [67].

Lemmatization is another pre-processing. In lemmatization, the word's suffix is either deleted or replaced, to return it back to its root called a lemma. Unlike stemmed words, a lemma is always a meaningful word [65]. The difference is that stemming only modifies words with a root word, while lemmatization is more detailed and attempts to replace words with another dictionary word [44]. In NLP, lemmatization is a common text pre-processing procedure and achieved good results. For instance, the word "caring" will have the meaningful word "care" as its lemma [65].

Table (2.2) Shortcut and Slang Words with Corresponding Word [66].

Shortcut	Corresponding word	shortcut	Corresponding word
ain't	is not	i'll	I will
amn't	am not	i'm	I am
aren't	are not	i'm'a	I am about to
can't	cannot	i'm'o	I am going to
'cause	because	isn't	is not
couldn't	could not	it'd	it would
couldn't've	could not have	it'll	it will
could've	could have	it's	it is
daren't	dare not	i've	I have
daresn't	dare not	kinda	kind of
dasn't	dare not	let's	let us
didn't	did not	mayn't	may not
doesn't	does not	may've	may have
don't	do not	mightn't	might not
e'er	ever	might've	might have
em	them	mustn't	must not
everyone's	everyone is	mustn't've	must not have
finna	fixing to	must've	must have
gimme	give me	needn't	need not
gonna	going to	ne'er	never
gon't	go not	o'	of
gotta	got to	o'er	over
hadn't	had not	ol'	old
hasn't	has not	oughtn't	ought not
haven't	have not	shalln't	shall not
he'd	he would	shan't	shall not
he'll	he will	she'd	she would
he's	he is	she'll	she will
he've	he have	she's	she is
how'd	how would	shouldn't	should not
how'll	how will	shouldn't've	should not have
how're	how are	should've	should have
how's	how is	somebody's	somebody is
i'd	I would	someone's	someone is

Shortcut	Corresponding word	Shortcut	Corresponding word
something's	something is	who'd've	who would have
that'd	that would	who'll	who will
that'll	that will	who're	who are
that're	that are	who's	who is
that's	that is	who've	who have
there'd	there would	why'd	why did
there'll	there will	why're	why are
there're	there are	why's	why is
there's	there is	won't	will not
these're	these are	wouldn't	would not
they'd	they would	would've	would have
they'll	they will	y'all	you all
they're	they are	you'd	you would
they've	they have	you'll	you will
this's	this is	you're	you are
those're	those are	you've	you have
'tis	it is	Whatcha	What are you
'twas	it was	luv	love
wanna	want to	sux	sucks
wasn't	was not	okay	ok
we'd	we would	idk	i do not know
we'd've	we would have	lol	laugh
we'll	we will	where'd	where did
we're	we are	where're	where are
weren't	were not	where's	where is
we've	we have	where've	where have
what'd	what did	which's	which is
what'll	what will	who'd	who would
what're	what are		
what's	what is		
what've	what have		
when's	when is		

2.5.2 Feature Extraction

After removing any unwanted data from the text, cannot fed the text directly to machine learning model because machine understands numerical data only. Features extraction is a process that extracts features from text and represented them as a numeric vector [40, 65]. The main objective of feature extraction is to minimize the dimensionality and eliminate irrelevant features and this will improve the performance of classification algorithm and reducing the time [68].

Traditional feature extraction techniques need handcrafted features. It takes time to create an effective feature. Recently, deep learning is consider a new feature extraction method and has made progress in text mining. The significant difference between deep learning and conventional methods is that deep learning learns features automatically from huge data rather than using handcrafted features that depend on the past knowledge of designer and not taking advantage of big data [57].

There are various text feature extraction techniques such as BOW, TF-IDF, one hot, word2vec, and count vectors.

2.5.2.1 Term Frequency Inverse Document Frequency (TF-IDF)

Is a widely used technique to evaluate the significance of a word in a document. Term frequency (TF) of a specific term (t) is computed as the number of times a term (t) appears in a document(d) divided by the number of words in the document. IDF is utilized to determine a term's importance [69]. Equation 2.7 [69] shows how to compute IDF.

$$IDF\ (t) = log\ [N / DF] \qquad (2.7)$$

Where N is number of documents and DF is number of documents that included term (t).

TF_IDF is calculated by Equation 2.8 [69].

$$TF\_IDF\ (t) = TF(t) * IDF(t) \qquad (2.8)$$

Term frequency is relatively similar to the BoW method. The text from the data is represented by term frequency as a matrix where rows are the total number of documents and columns are the total number of distinct words used in all of the documents. BoW ignores the order of words in documents and grammar [70].

words with a large TF-IDF value are more significant than terms with a small TF-IDF value [70].

2.5.2.2 Word2vec

Word2vec is one of the most word embedding techniques proposed by Google in 2013. Word2vec is a neural network model. There are two learning models continuous bag of words (CBOW) and skip-gram. CBOW predicts the term from the context while skip-gram predicts the context from the term [71]. With a huge corpus of text as input, word2vec creates a vector space, generally with several hundred dimensions, and assigns each distinct word in the corpus a corresponding vector in the space. Word vectors are positioned in the vector space so that words with similar semantic and syntactic properties are near one another in the space, while terms that are less similar are placed apart to one another [72]. Word2Vec's weight is determined by word order or location rather than the frequency in the same context. The similarity between the two words can be assessed after the weight of Word2Vec has been determined. Word2Vec can represent each word as a low dimensional vector and this makes adding new words to the vocabulary list simple and easy to incorporate new sentences [70].

A portion of the Google news dataset (about 100 billion words) was used to train pre-trained vectors. For 3 million words and phrases, the model has 300-dimensional vectors [73]. This thesis is used pre-trained vectors from Google news.

2.6 Evaluation

Evaluation is used to evaluate the performance of classification. This thesis used confusion matrix. A confusion matrix is a method for predictive analysis in machine learning. It evaluates the performance of classification for the model. In addition, the confusion matrix can be thought of as a summary table of the number

of correct and incorrect predictions produced by a classifier (or classification model) for binary classification tasks [74].

A confusion matrix is an N x N matrix which utilized to assess the effectiveness of a classification model, where N is the total number of target classes. By seeing the confusion matrix, a person could identify the model's accuracy by observing the diagonal values for measuring the number of accurate classifications [74]. The confusion matrix is a square matrix, where each column represents the model's predicted value while each row contains the actual values [75]. Figure (2.8) shows confusion matrix.

		Assigned Class	
		Positive	Negative
Actual Class	Positive	TP	FN
	Negative	FP	TN

Figure (2.8) Confusion matrix [75]

True Positive (TP): means that both the actual and predicted values were positive.

False Positive (FP): Despite the actual was negative, the prediction was positive.

False Negative (FN): the prediction is a negative and the actual value was positive.

True Negative (TN): The predicted value was negative and the actual value was negative.

This thesis used four metrics to evaluate the performance of the model. Accuracy is the number of samples that are predicated correctly by the model divided by the total number of predictions generated by the model. It is calculated by Equation 2.9 [25].

$$\text{Accuracy} = \frac{Tp+Tn}{Tp + Fp + Fn + Tn} \quad (2.9)$$

Precision represents the number of tweets that are classified as depressed and are truly predicted to be depressed. It is computed by Equation 2.10 [25].

$$\text{Precision} = \frac{Tp}{Tp+Fp} \quad (2.10)$$

Recall is the number of depressed tweets that are exactly predicated by the model out of all the positive examples. Equation 2.11 explains how to find it [25].

$$\text{Recall} = \frac{Tp}{Tp+Fn} \quad (2.11)$$

F1-Measure is explicated as a symmetric average of the precision and recall. Equation 2.12 shows how it was computed [25].

$$\text{F1-Measure} = 2* \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (2.12)$$

Chapter Three

The Proposed System

CHAPTER THREE

THE PROPOSED SYSTEM

3.1. Introduction

This chapter involves, the main stages of the proposed system used in this thesis. Firstly, the architecture of the proposed system is presented. Then, is followed by a description of the dataset and the preprocessing steps that were used. Features extraction methods that are applied. Finally, the classification techniques that are performed will be illustrated.

3.2 The Proposed System Architecture

The proposed system includes four stages. Each stage consists of many steps to achieve the objectives of the thesis. These stages are: preprocessing, features extraction, classification, and evaluation.

The first stage is preprocessing and this stage contains different steps to clean and prepare data for the next stage. The second stage extracts feature from text using TF-IDF and word2vec methods. The third stage is a classification that applied various machine and deep learning techniques to identify if the tweet is depressed or not. The fourth stage is an evaluation to evaluate the performance of the model. In this thesis suggested two proposed systems. The first system for traditional machine learning. the second is for hybrid model. Figures (3.1) and (3.2) presents the two systems.

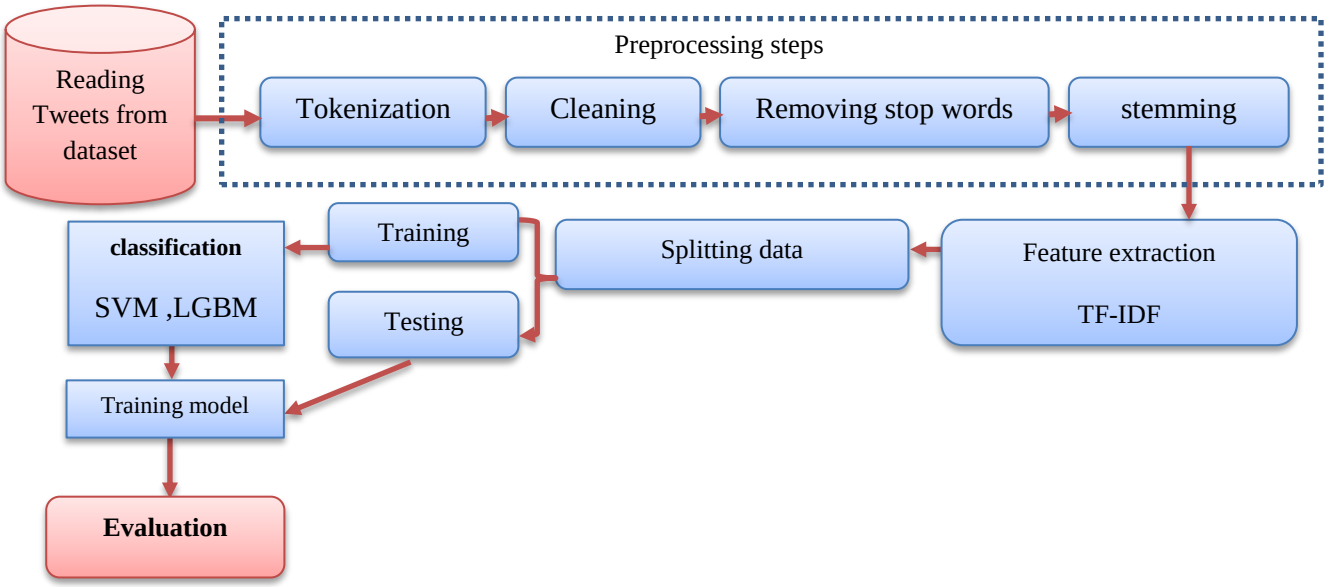


Figure (3.1) The Proposed System Architecture for Traditional Machine Learning.

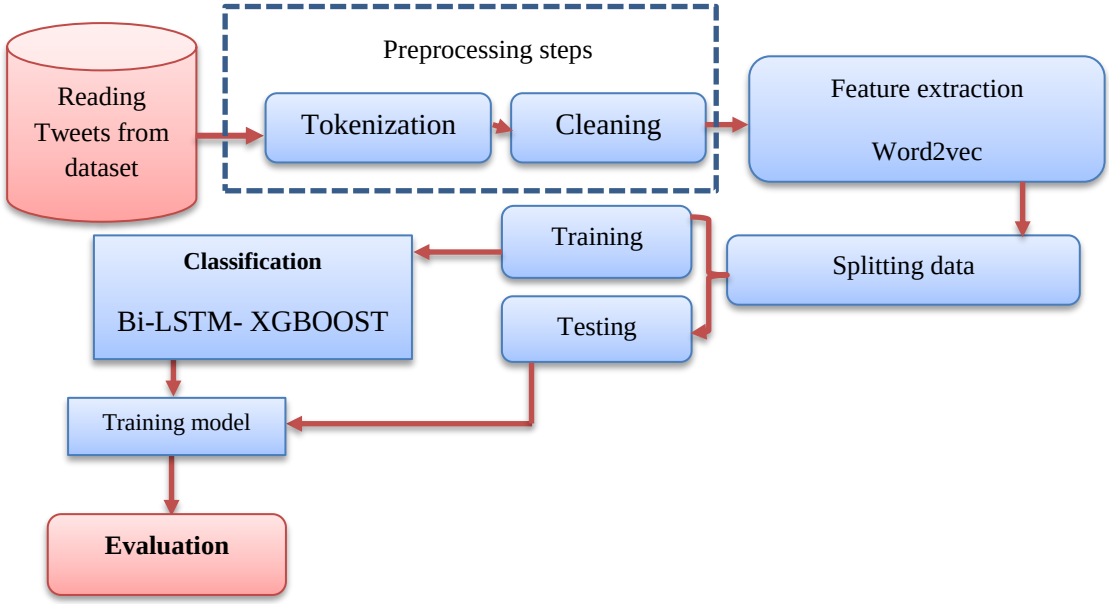


Figure (3.2) The Proposed System Architecture for Hybrid Model.

3.2.1 Dataset

As the previous mention in chapter two, this thesis used two datasets from Twitter. The first dataset contains more than 7000 tweets splitting into two comma separated values (CSV) files, one for depressed tweets and the other for non-

depressed tweets. In addition, there are 32 columns, and just two columns are used and it is a tweet and label.

Two reasons lead to use these datasets:

1. The datasets are public and available on the Kaggle website. These data were also adopted in previous studies
2. Using more than one dataset to find out the efficiency of the proposed system.

3.2.2 Preprocessing

Tweets are written by people, thus they may include slang words, acronyms, emoticons, or unintelligible symbols. Therefore, texts must be processed before extraction features and classification. Preprocessing is the process of cleaning raw data to become appropriate for machine learning classifiers. Algorithm (3.1) explains these steps. The following steps explains the traditional preprocessing and the new preprocessing in details:

A. Cleaning the Dataset from Irrelevant Tweets

When checking the tweets in the first dataset notice that some tweets contain only URL links for images or irrelevant subjects. Thus, before beginning preprocessing and other processes must ensure that all tweets contain real data and remove all irrelevant tweets. Step 1 in Algorithm (3.1) shows this step.

B. Tokenization

Tokenization is one of the important processes in NLP. It splits the text into smaller parts called the tokens. Therefore tokenization is applied here to splitting each tweet into individual words. The splitting based on space between words, To make it easier to handle in preprocessing. Step 2.1 in Algorithm (3.1) illustrates it.

C. Replacing Emoticon and Emoji with Correspond Text

The emoji or emoticon is used to express feelings, such as sadness or happiness, thus it is necessary to keep them in text and not remove them from text, they would

be replaced with the corresponding text. For example the emoticon (-: becomes smiley and)-: becomes sad, and emoji 💕 becomes two hearts. This pre-processing step is ignored in previous studies. Steps 2.2 and 2.3 in algorithm (3.1) states it. In addition, Table (2.1) in chapter two explains the emoticon and correspond text. While there are function to convert emoji in python**.**

Algorithm (3.1) preprocessing of Tweets

Input : DF-tweets : data frame contains tweets.
Output: DF-clean-tweets is a data frame that contains the tweets after preprocessing.

Begin
Step 1 : cleaning the dataset from irrelevant tweets.
Step 2 : cleaning the tweets from unwanted symbols or words.
For each tweet in DF-tweets do
Step 2.1: splitting tweets into tokens.
Step 2.2 : replacing emoticon with correspond text.
Step 2.3: replacing emoji with correspond text.
Step 2.4: removing URL links, mentions, and hashtag symbol.
Step 2.5:removing all characters expect alphabets, apostrophes, and single quotation mark.
Step 2.6: replacing apostrophes with single quotation mark.
Step 2.7: replacing slang and shortcut words with stander word
Step 2.8: removing single quotation mark for non-apostrophes character.
Step 2.9:converting upper case to lower case.
DF-clean-tweets = tweets after cleaning
End for
Step 3: removing stop words from tweets and the word with lengths less than 2 letter.
DF-clean-tweets=tweets after removing stop words.
Step 4: stemming or lemmatization
Step 4.1: stemming
Step 4.2: lemmatization
DF-clean-tweets=tweets after stemming or lemmatization.
End

D. Removing URL Links, Mentions, Hashtag Symbols, and HTML Tags

The tweets may be contain URL links for images or pages and this is not useful through analysis of tweets. In addition, tweets may include mentions for other users , hashtag symbols, or HTML tags. Those not necessary in analysis of tweets. Therefore, it must remove them. Thus, this thesis remove it. Step 2.4 in Algorithm (3.1) shows it.

E. Removing Punctuations

One of the most important steps in text preprocessing is the removal of useless information, such as punctuation and special characters. Punctuation illustrates how a sentence is constructed, how it should be read, and how it should be understood.

Punctuation like a question mark ‘?’, comma ‘,’ or apostrophe ‘ ‘ ’ and semicolon ‘;’. In addition, there are special characters such as &,*,%,=, and $. All these will be deleted from tweets in steps 2.5 and 2.8 in Algorithm (3.1). In step 2.5 deleting all special characters except alphabet and apostrophe or single quotation mark because shortcut words such as ‘I'm’ includes an apostrophe. Thus, it must not be removed before replacing it with a stander word in step 2.7. The shortcut words may contain apostrophes or single quotations. Therefore, step 2.6 will replace apostrophes with a single quotation. After replacing them with stander words in 2.7, will remove them in step 2.8.

F. Replacing Slang and Shortcut Words with Corresponding Text

When looking at the most frequent depressing words in Figures (3.3) and (3.4), notice that the depressed users widely use words like ('ok’, ’want’, ’never’, ’cannot’). Moreover, in tweets, some users use slang words like ‘okay’, ’wanna’, ‘ne'er’ and ‘can't‘ instead of ‘ok’, ‘want’, ‘never’, and ‘cannot’ respectively as shown in Figure (3.3). Therefore, these words must be replaced with stander words

to be in one form. This pre-processing step is ignored in previous studies. Step 2.7 will do that in Algorithm (3.1). Table (2.2) in chapter two involves the shortcuts and slang words with corresponding words.

G. Converting Upper Case to Lower Case

The text may involve the same words but one of them starts in upper case and the other in lower case, in this step will convert all letters to lower case to solve this problem. Step 2.9 in Algorithm (3.1) includes this process.

```
Tweet 1:“are you okay?” me: “no, but it’s okay.”
Tweet after replacing with stander word :are you ok me no but it
is ok


Tweet 2:Feel like shit lol
Tweet after replacing with stander word: feel like shit laugh


Tweet 3:actually just wanna end it all. idk what keeps me here.
Tweet after replacing with stander word: actually just want to end
it
all i do not know what keeps me here

Tweet 4:no i can't just "treat myself in a kind and
loving way".
Tweet after replacing with stander word: no i cannot just treat my
self in a kind and loving way
```

Figure (3.3) Example from Depressing Tweets in the Dataset show Shortcut Words and Slang Words.

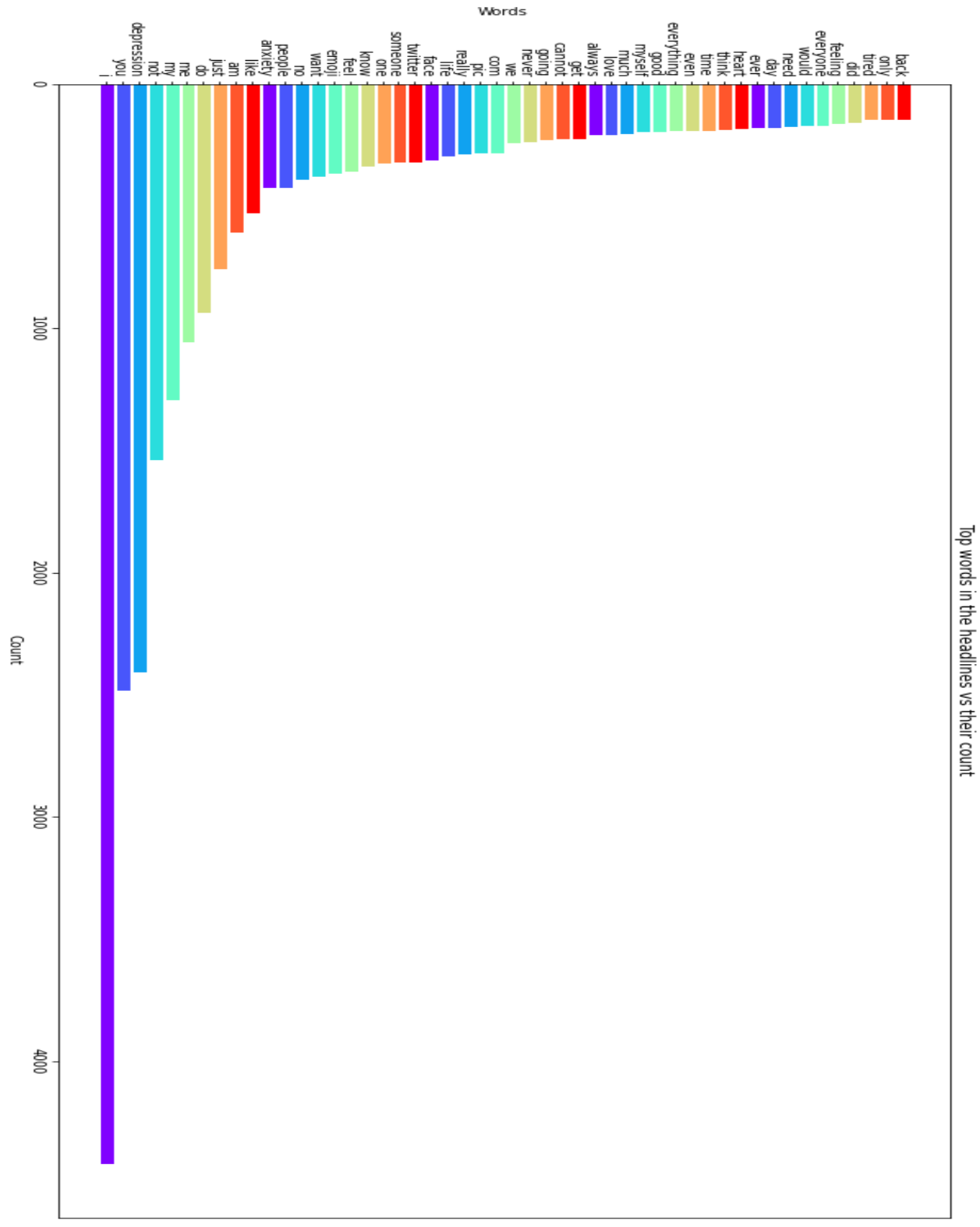


Figure (3.4) Most Fifty Frequent Depressed Words that extracted from the First Dataset by using a Word Cloud.

H. Removing Stop Words from Tweet

Step 3 in Algorithm (3.1) shows removing stop words from tweets. In this thesis there are two methods were used to delete stop words. First, the traditional method removes all stop words. The Second method suggested keeping some important stop words such as first or second-person pronouns (‘I’, ‘my’, ’myself’, ’you’) (see Figures 3.5 and 3.6). There is some research that noticed that most depressed persons tend to use these words more than other people to express themselves [23,78,79].

In addition, in this thesis and through viewing the words cloud (see Figure 3.4) and most depressed tweets in the datasets (see Figure 3.3) notice that depressed people widely used negation words such as (no, not) and these words changed the meaning of the sentence from a natural sentence to a sad sentence when coming with positive words such as ‘I am not happy.' Thus, this thesis proposed keeping these words and not removing them from tweets. Figure (3.6) shows the proposed stop words list that not remove and called customized removing stop words list.

All stop words
'a', 'about', 'above', 'after', 'again', 'against', 'ain', 'all', 'am', 'an', 'and', 'any', 'are', 'aren', "aren't", 'as', 'at', 'be', 'because', 'been', 'before', 'being', 'below', 'between', 'both', 'but', 'by', 'can','couldn', "couldn't", 'd', 'did','didn',"didn't", 'do','does','doesn',"doesn't", 'doing', 'don', "don't", 'down', 'during', 'each', 'few', 'for', 'from','further','had','hadn',"hadn't",'has','hasn', "hasn't", 'have', 'haven', "haven't", 'having', 'he', 'her', 'here', 'hers', 'herself', 'him', 'himself', 'his', 'how','i','if', 'in', 'into', 'is','isn', "isn't", 'it', "it's", 'its', 'itself', 'just','ll','m','ma','me','mightn',"mightn't", 'more','most','mustn',"mustn't", 'my','myself','needn', "needn't", 'no', 'nor', 'not', 'now', 'o', 'of', 'off', 'on', 'once', 'only', 'or', 'other', 'our', 'ours', 'ourselves', 'out', 'over', 'own', 're', 's', 'same','shan', "shan't", 'she', "she's", 'should', "should've",'shouldn',"shouldn't", 'so', 'some', 'such', 't', 'than', 'that', "that'll", 'the', 'their', 'theirs', 'them', 'themselves', 'then', 'there', 'these', 'they', 'this', 'those', 'through', 'to', 'too', 'under', 'until', 'up', 've','very', 'was','wasn',"wasn't", 'we','were','weren',"weren't", 'what', 'when', 'where', 'which', 'while', 'who', 'whom', 'why', 'will', 'with', 'won', "won't",'wouldn', "wouldn't", 'y', 'you', "you'd","you'll","you're", "you've", 'your', 'yours', 'yourself', 'yourselves'.

Figure (3.5) All Stop Words

customized removing stop words list
"i", "me", "my", "myself", "mine", "we", "our", "ours", "ourselves", "yourself", "you", "am", "no", "not", "only", "does", "do", "did", "own", "cannot", "could not", "should", "just", "must", "could"

Figure (3.6) Customized Removing Stop Words List.

I. Stemming

This process is a very important step in reducing the number of words that have the same meaning by removing the affixes from the end of the word. For example, stemming reduces the words (likes, liked, likely, liking) to root word 'like'. The stemming method that is used in this thesis is Porter Stemmer. Step 4.1 represent stemming process in Algorithm (3.1).

J. Lemmatization

Is the process of collecting the various forms of a word to analyse it as a single element. It is the same stemming but it focuses on the meaning of the words. Step 4.2 represent lemmatization process in Algorithm (3.1).

3.2.3 Features Extraction

Machine learning algorithms are unable to understand tweets as well as people. Therefore, it must convert the tweets into a format that machine learning can understand. In addition to extracting the most significant features in the text, features extraction methods are utilized to extract features in a numerical form that machines can understand . Two common features extraction were performed in this thesis. TF-IDF and word2vec.

3.2.3.1 TF-IDF

One of the most features extraction methods from text. This technique computes the importance of word by count the times that the word appear in the all documents in the document set. After determining the TF and IDF values, it can

compute the TF-IDF. The Equations (2.7, 2.8) in Chapter Two explain how to compute TF-IDF.

A. Creating Vocabularies

The first step is created a list of unique terms from text in training set called vocabulary. The words in each tweet are chosen without repetition. To determine the frequency of each word, the words which resulted from the original text were ordered alphabetically. A list of indexes were created from all words. Each feature has two elements (key, token) key represents the index of word in the list and token act the feature. The below example in Figure (3.7) explains how to create the vocabularies.

vocabularies
Tweet 1: i cannot make you understand. *Tweet 2: i cannot make anyone understand what is happening inside me.* *Tweet 3: i cannot even explain it to myself.* *Vocabularies ={cannot: 1, make: 6, you: 10, understand: 9, anyone: 0, happening: 4, inside: 5 , me: 7, even: 2, explain: 3, myself: 8}*

Figure (3.7) How to create the vocabularies

B. Creating a Vector of Features

The text is represented as vector space where each text is a matrix of features (terms). The dimensions or number of columns match the total number of features for all texts. The vector's length determines depending on the length of the feature, therefore it is the same for all texts. To prevent needlessly big features, Tokens are only considered as features if they occur more frequently than other features. Then, each text will compare with the feature and compute the number of times for the appearance of the feature. Additionally, each feature has a weight that

indicates how significant it is to the document. Text is represents as (w_1, w_2, w_3,…w_n) where w_i is the weight for feature i in text T.

C. Computing Features Weight

After created feature vector for text, now can computing the weight for each feature using TF that refers to term frequency and IDF represents inverse document frequency. Table (3.1) shows how to compute the weight.

Table (3.1) How to Compute the Weight

	0	1	2	3	4	5	6	7	8	9	10
Tweet 1	0	0.3731	0	0	0	0	0.4804	0	0	0.4804	0.6317
Tweet2	0.4261	0.2517	0	0	0.4261	0.4261	0.3241	0.4261	0	0.3241	0
Tweet3	0	0.3227	0.5464	0.5464	0	0	0	0	0.5464	0	0

3.2.3.2 Word2vec

Is one of the most embedding techniques. Word2vec created the words vectors. The first step is build a vocabulary from input text. For each word there are vector. To create this vector we require a set of data indicating the terms that frequently appear near a specific word. To accomplish this, we'll use a tool called a context window. For example "Sleep be so hard when you cannot stop think," (see Figure 3.8). Therefore, must define something called window size. In this instance, let's use 2. We repeat over all the words in the given data, which in this example is only one sentence. Since our window size in this case is 2, we will take into account the two words that come before and after each word, giving each word a total of four words that are related to it. We will repeat this process for each and every word in the data.

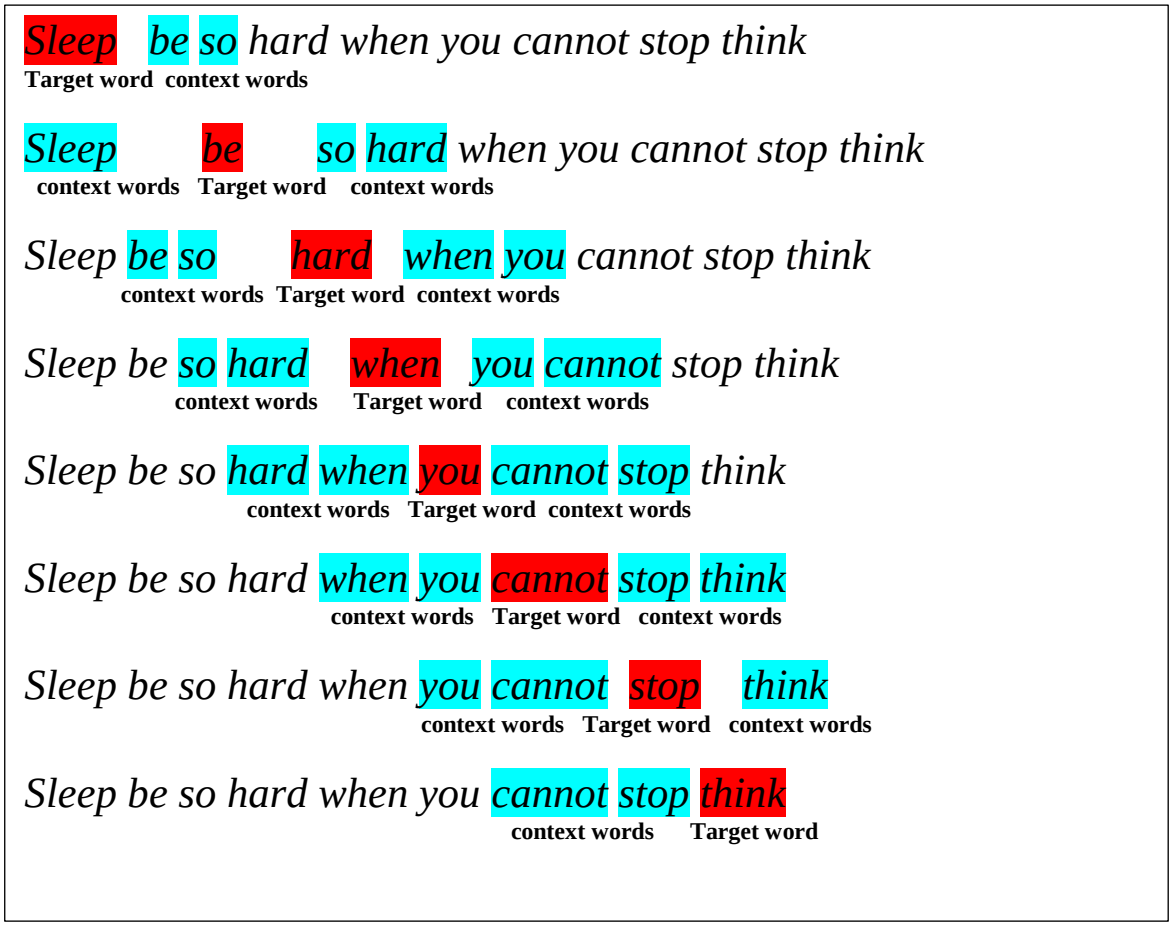


Figure (3.8) Context Window iterate Over all the Words in the Given Sentence.

Finding all pairs of target and context words allows us to create a dataset in the format of target word and context word as we pass the context window across the text data. Thus forth, final target word vs. context word data set will be as in Figure (3.9).

(sleep, be),(sleep, so), (be, sleep), (be, so), (be, hard), (hard, be), (hard, so), (hard, when) ,(hard, you), (when, so), (when, hard), (when, you), (when, cannot), (you, hard),(you, when), (you, cannot), (you, stop), (cannot, when), (cannot, you), (cannot, stop), (cannot, think) ,(stop, you), (stop, cannot), (stop, think), (think, cannot).

Figure (3.9) All Pairs of Target and Context Words for the Sentence.

This can be viewed as "training data" for word2vec. Neural network is predicate the target words or context words based on the type of architecture. There are two type CBOW and skip gram. Using a target word as a starting point, the skip gram model attempts to predict each context word. While CBOW is predict a target word from context word. After creating the vector for each word in sentence, they sum values in one vector. Simply look at the diagram below (Figure 3.10) as an instance. One sentence is utilized to illustrate the flow; this occurs for each sentence in the corpus. A single vector is created by adding all of the vectors. The information contained in the sentence is represented by a single vector, hence it is considered as one row. This thesis used the pre-trained google news word2vec.

3.2.4 Splitting Data

Before classifying the tweets by classifier models it must splitting data into t raining and testing data. Thus, after cleaning data from Irrelevant tweets, the first d ata set has 10264 tweets divided into 5571 depressed tweets and 4693 non-depress ed tweets. While the second dataset remains as it, where there are 62,000 tweets. It contains 32,000 depressed tweets and 30,000 non-depressed tweets.

With traditional machine learning techniques, this thesis spilt data as this : 80 % for training the models and 20% for testing. The first dataset splitting into 8211 tweets for training and 2053 for testing. The second dataset is includes 49600 tweets for training and 12400 tweets for testing.

While, splitting data with deep learning like this: the details of splitting in firs t dataset is 85% for training and 10% from training for validation and 15% for testin g. Thus, the first data become after splitting 7851 tweets for training,873 tweets for validation and 1540 tweets for testing. The details of splitting in second dataset is 80% for training and 30% from training for validation and 20% for testing. The second data divided into (training: 34720, validation: 14,880, testing: 12400).

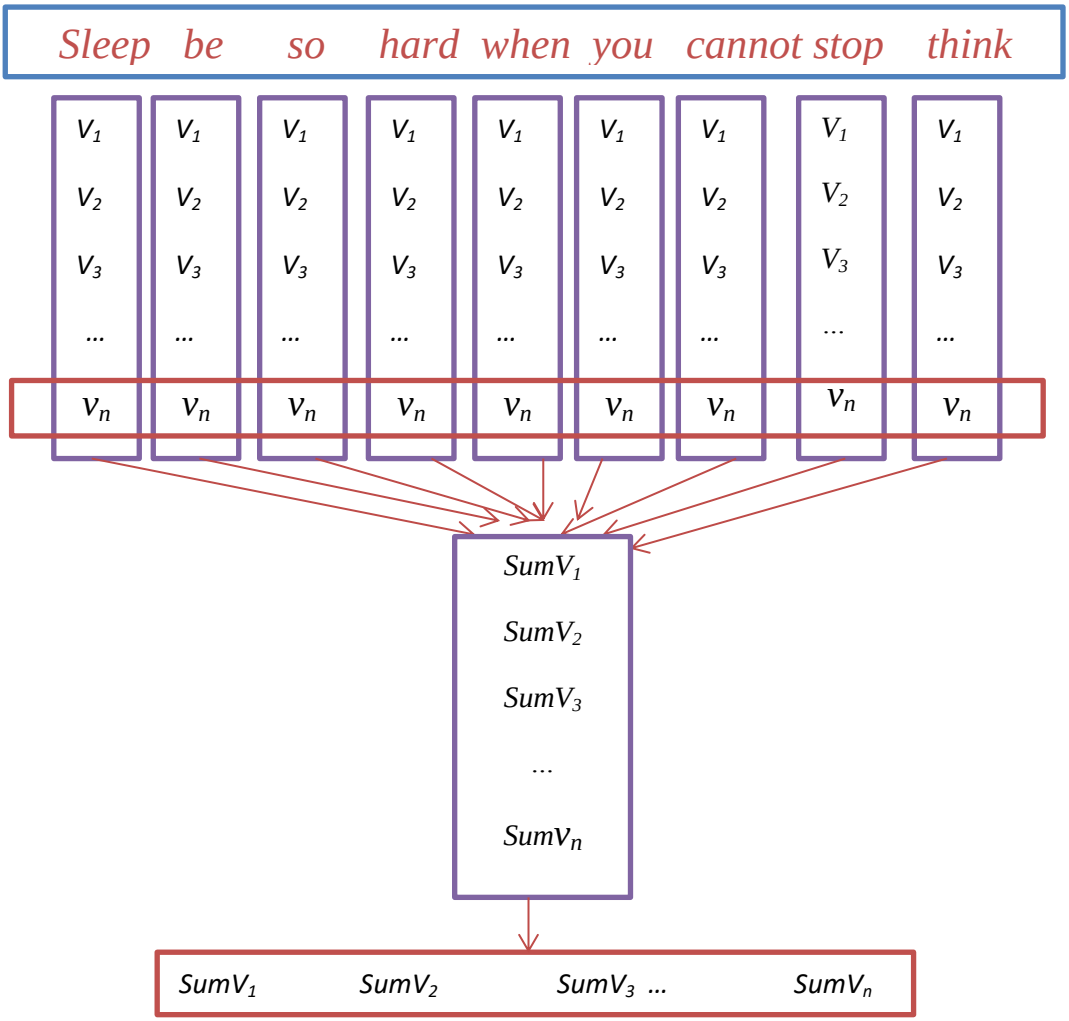


Figure (3.10) How to Create Vector Word in the Word2vec Method.

3.2.5 Classification Methods

Classification is one of most significant process .It is the process of classifying objects into one of many classes. This thesis used classification techniques to classify the tweets to depressed tweet with label (1) or non-depressed tweet (0).

Three types of machine learning techniques were used and involves :

1) Traditional machine learning techniques :
   (SVM, LR, LGBM, DT, XGBOOST).

2) Deep learning techniques and includes Bi-LSTM and CNN. Will explains more in the next sections.
3) Hybrid deep learning and machine learning techniques. It will also be explained in the next sections.

3.2.5.1 Traditional machine learning techniques

Five techniques were used SVM, LGBM, DT, LR, and XGBOOST.

3.2.5.2 Deep Learning Techniques

Two deep learning techniques were applied in this thesis Bi-LSTM and CNN.

A. Bi-LSTM

The first deep learning technique is Bi-LSTM. This method was applied for its ability to analyze sequential data such as texts and extract features. In addition, its ability to keep previous and future information because it has two LSTM layers forward and backward layers. The tweets before classifying by deep learning technique are preprocessed then after that, the words are converted to numbers then, the length of tweets will be equal by padding sequence. The proposed Bi-LSTM includes a set of layers. Figure (3.11) and Table (3.2) show the details of Bi-LSTM.

- Embedding layer : it used word2vec matrix as weight for training.
- Bi-LSTM layer: it contains two LSTM layer. The first layer is forward LSTM. It read the sentence from left to right. The second layer is backward and it read sentence from right to left.
- Max pooling layer: extract important features from the features map and ignore other information.
- Dense layer: it works as an output layer with an activation function. In this thesis two activation functions Relu and sigmoid. Relu is used to convert the negative value to 0 and sigmoid to classify binary values.

Dropout layer: it is used to reduce overfitting in the model.

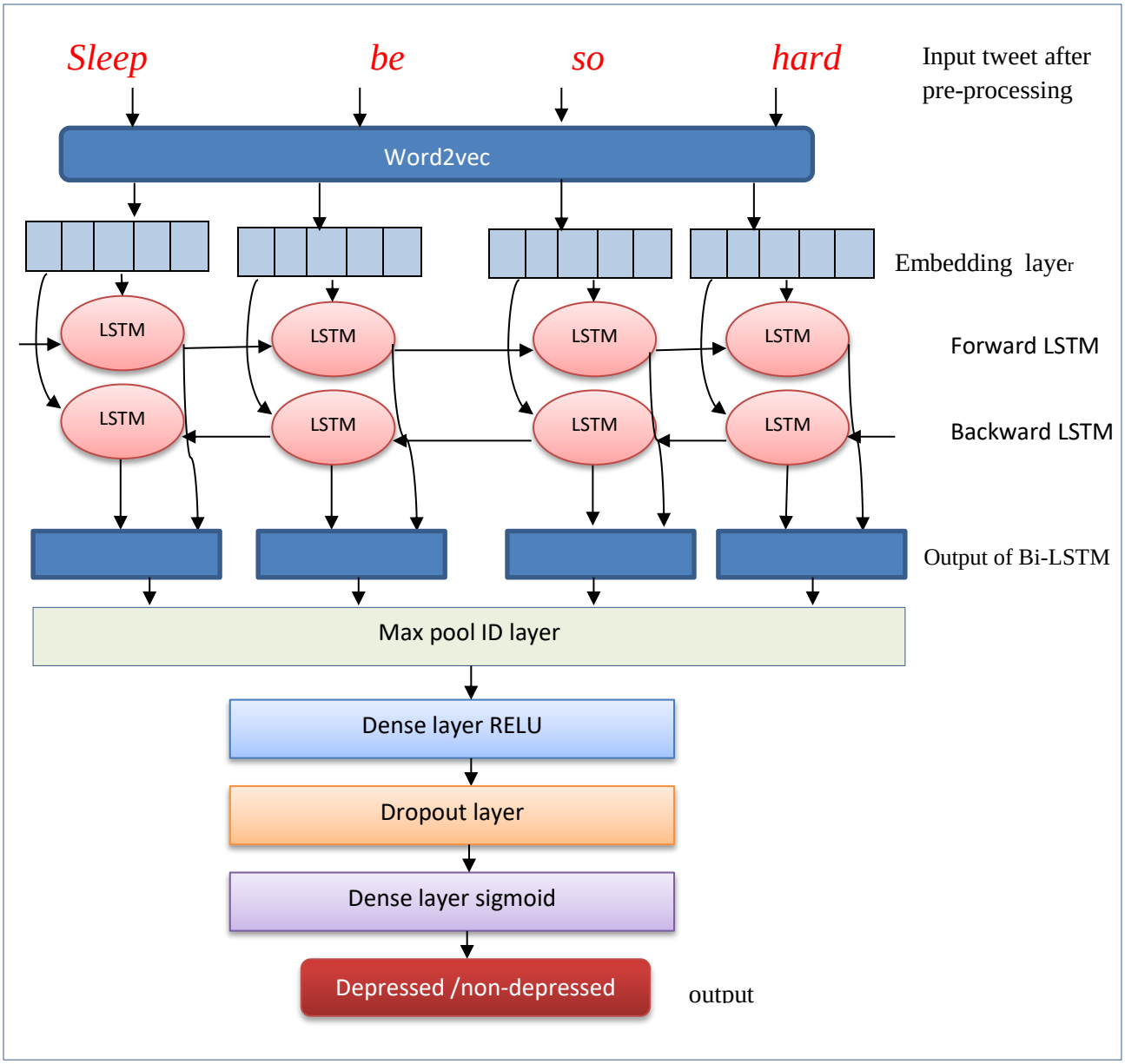


Figure (3.11) Bi-LSTM Model.

Table (3.2) The parameters for the Bi-LSTM layers

Layer	The parameters for the layer
Input	max length sequence =140
Embedding	Max number words = 7000, Embedding dimension =300, Weights = matrix-word2vec.
Bidirectional	LSTM units =300, Dropout = 0.5
GlobalMaxPool1D	-
Dense	units =300, activation=Relu , regularizes l2=0.01
Dropout	Dropout = 0.5
Dense	Activation = sigmoid

B. CNN

CNN is one of the most common techniques used to classify images but nowadays is widely used to classify text. It is characterized by its ability to extract features from the text and speed. The proposed CNN model has the following layers:

- Embedding layer : it used word2vec matrix as weight for training.
- Convolutional layer: it pass filters on embedding matrix to create a features map.
- Max pooling layer: extract important features from the features map and ignore other information.
- Flatten layer: it converts 2D array to single vector.
- Dense layer: it works as an output layer with an activation function. In this thesis two activation functions RELU and sigmoid. RELU is used to convert the negative value to 0 and sigmoid to classify binary values.
- Dropout layer: it is used to reduce overfitting in the model. Figure (3.12) and Table (3.3) show the CNN model.

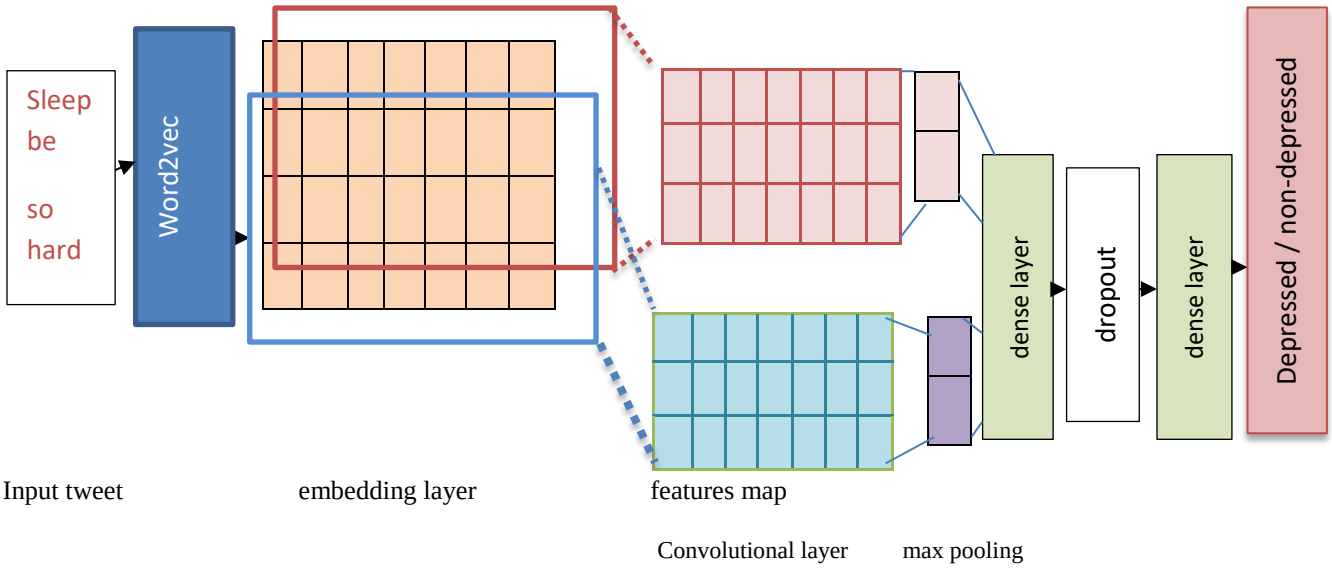


Figure (3.12) Explains the Details of CNN Model.

Table (3.3) The parameters for the CNN layers

Layer	The parameters for the layer
Embedding layer	Max number words = 7000, Embedding dimension=300, Weights = matrix_word2vec , max length sequence =140
Convolutional layer	filters =32, padding=same ,stride =3, activation=Relu
MaxPooling1D layer	-
Flatten layer	-
Dense	units =10, activation=Relu , regularizes l2=0.01
Dropout	Dropout = 0.5
Dense	Activation = sigmoid

3.2.5.3 Hybrid Models

This thesis suggests a hybrid model that combines features of deep learning and machine learning. The proposed model is divided into two parts. The first part is Bi-LSTM or CNN which extracts features and information from input text. The second part is replacing the last layer (dense (sigmoid)) from CNN or Bi-LSTM with traditional machine learning and classifying it. Three classifiers suggested SVM, LGBM, and XGBOOST. Figures (3.13), and (3.14) explain the details of the models.

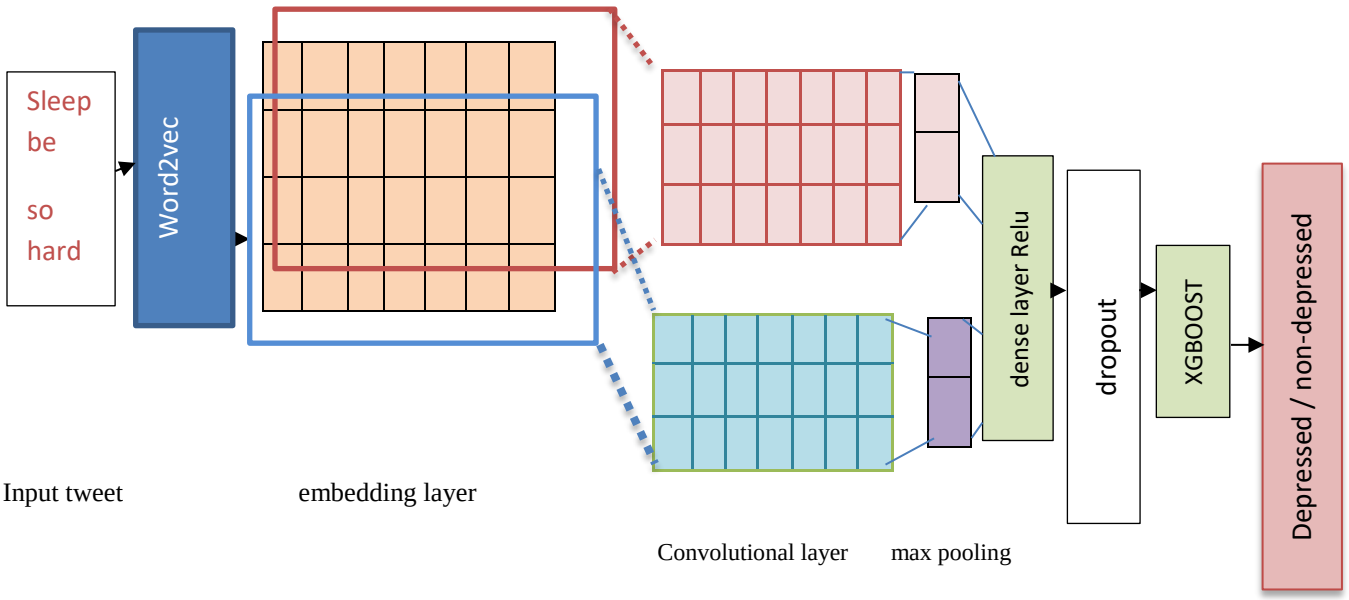


Figure (3.13) Hybrid Model CNN -XGBOOST.

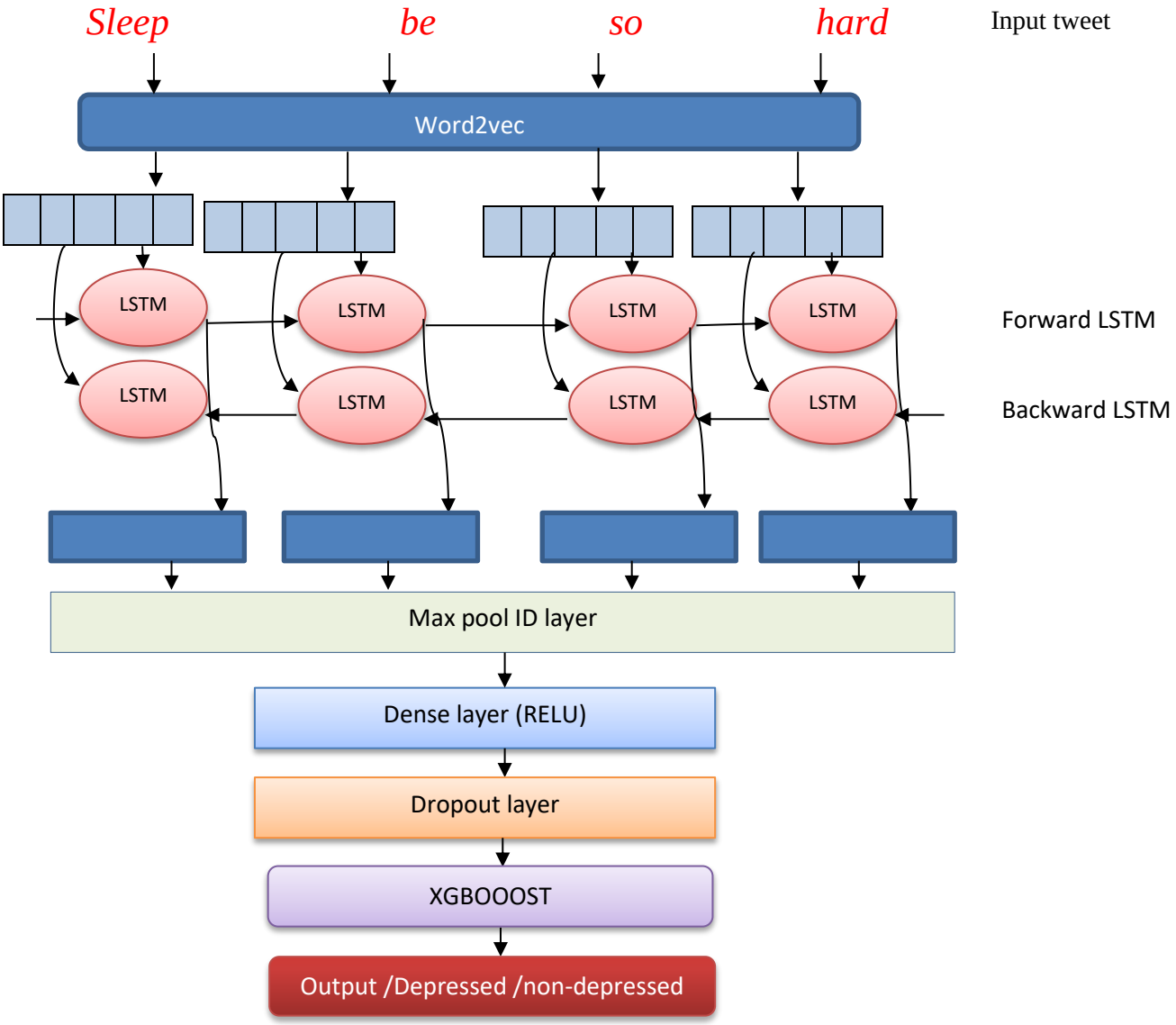


Figure (3.14) Hybrid Model Bi-LSTM -XGBOOST.

Chapter Four

Experimental Results and Discussion

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Introduction

The results of each stage for the proposed system as described in Chapter Three are discussed in this chapter. The outcomes of each stage are arranged according to where they appear in Chapter Three. However, The hardware and software requirements for implementing the suggested system, come first in this Chapter.

4.2 Software and Hardware

The proposed system was implemented using the following hardware and software requirements.

Hardware

Processor Intel i7, RAM 8GB, Storage 320 GB, Freq.2.7GHz, Memory: 4096MB RAM.

Software: Operating System: Windows10 -pro-64-bit.

Programming language: Python language.

IDE: the system was implemented by Python 3.9.12, Jupyter Notebook.

4.3 Dataset

This thesis used two public datasets from the Kaggle website. The first dataset [26] was created by Hyun Ki Cbo in 2021. It was gathered quickly from Twitter to work with machine learning techniques. It has two file one for depressed tweets and other for non-depressed. A dataset containing more than 7000 tweets that belongs to depressed and not depressed users. The depressed file contains 3496 tweets and non-depressed file involve 4809 tweets. It includes 38 columns. The dataset is imbalanced data. 2313 depressed tweets [76] were added for this dataset to improve accuracy and increase the number of

depressed tweets. The number of depressed tweets become is 5,809 and the total number is 10,618. Table (4.1) explains some important columns in dataset.

Table (4.1) Some Important Columns in Dataset [26]

No	Name of column	Description
1	Id	The unique Number for tweet
2	Date	The date of tweet
3	Tweet	The content of tweet
4	User-id	The unique number for user
5	Language	Writing language for tweet
6	Username	The name of user that appear for people
7	Name	Original name for user
8	Day	Day of creation of the tweet
9	Hour	Hour of creation of the tweet

The second dataset [77] was collected by Samrat Sahoo in 2020 from Twitter. It contains two files, each file has 31,000 tweets, divided into 16000 depressed tweets and 15,000 non-depressed tweets. Furthermore, it includes two columns. There are two columns in this dataset tweets and label.

4.4 Results of Data Preprocessing

The results of the preprocessing stage are presented after applying tokenization to split tweets into words. It includes the following sub-steps:

- **Results of Replacing Emoticon and Emoji with Correspond Text**

In this step, each emoticon such as (-: will replace with correspond text. In addition, emoji such as will replace as shown in Figure 4.1.

Replacing emoji
Before replacing *I don't feel good....but like in a good way if that makes sense*  After replacing *i do not feel good but like in a good way if that makes sense grinning face with sweat face with tears of joy face with tears of joy*
Replacing emoticon
Before replacing *"oh they're just really sensitive" good job on invalidating someone's emotions :)* After replacing *oh they are just really sensitive good job on invalidating someone is emotions smiley*

Figure (4.1) Replacing Emoji and Emoticon with Correspond Text.

- **Removing URL Links**

This step includes removing URL links for websites or images from tweets. Figure (4.2) explains this step.

Removing URL
Before removing *12 years ago, I called my dad and he gave me some great advice. A short (1 min) story.. https://t.co/zjmfEzM6* After removing *years ago i called my dad and he gave me some great advice a short min story*

Figure (4.2) Before and After Removing URL Links.

- **Removing Mentions**

the tweet may be involve mentions **@username.** the outcomes of this step illustrated in Figure (4.3).

Removing mentions
Before removing *the best thing I heard this week: "I don't do public math" -@APompliano he's right. he's absolutely right. I stand for my right to not be forced to do public math* After removing *the best thing i heard this week i do not do public math he is right he is absolutely right i stand for my right to not be forced to do public math*

Figure (4.3) Before and After Removing Mentions.

- **Removing Hashtag**

Each hashtag symbol # will remove from a tweet. Figure (4.4) shows the results of removing hashtags.

Removing hashtag
Before removing *I love the #solidarityat8 idea* After removing *i love the solidarityat idea*

Figure (4.4) Before and After Removing Hashtag.

- **Removing Punctuations**

Removing all characters and symbols except apostrophes or single quotations ‘ as can see in Figure (4.5) where the apostrophe is kept in “I’m” and “you're”. The outcomes of this step are shown in Figure (4.5).

Removing punctuations
Before removing *when i say "i'm okay" i want someone to look me in the eyes, hug me tight, and s ay, "i know you're not."* After removing *when i say i'm okay i want someone to look me in the eyes hug me tight and say i know you're not*

Figure (4.5) Before and After Removing Punctuations.

- **Replacing Slang and Shortcut Words with Corresponding Text**

The slang and shortcut words will replace with stander words to be in one form. Figure (4.6) illustrated.

Replacing slang and shortcut words

Before replacing
"are you okay?" me: "no, but it's okay."

After replacing
are you ok me no but it s ok

Before replacing
typing 'haha' when you can't even smile, acting like you're happy when all you want to do is cry, tell everyone you're okay when you're not

After replacing
typing haha when you cannot even smile acting like you are happy when all you want to do is cry tell everyone you are ok when you are not

Figure (4.6) Replacing Slang and Shortcut Words.

- **Converting Upper Case to Lower Case**

Each upper case letter will convert to lower case. Figure (4.7) shows results of this step.

Converting upper case to lower case

Before converting
I just worked out how to take photo's on twitter jeeeppeeee! I know it took me a year!!!!!!!! No laughing please lol

After converting
i just worked out how to take photo s on twitter jeeppee i know it took me a year no laughing please laugh

Figure (4.7) Converting Upper Case to Lower Case.

- **Removing Stop Words from Tweet**

Remove all stop words that not important and keeping only words that widely used by depressed people. Figure (4.8) explains the results of this step.

Removing stop words from tweet
Before removing *it sucks, doesn't it? feeling like you're not good enough, no matter how hard you try.* After removing *sucks does not feeling like you not good enough no matter hard you try*

Figure (4.8) Removing Stop Words from Tweet.

- **Stemming**

This means removing prefixes and suffixes from words. This step is important because of reducing the number of words. Figure (4.9) shows the results of stemming for tweet.

Stemming
Before stemming *my biggest mistake ever is thinking that people care for me as much as i do for them, but in reality it's almost always one sided.* After stemming *my biggest mistak ever think peopl care me much i do realiti almost alway one side*

Figure (4.9) Stemming Process.

- **Lemmatization**

Returning the words to their origin with keeping their meaning by comparing with their own dictionary. Figure (4.10) explains the results of lemmatization.

Lemmatization

Before lemmatization

i don't tweet for sympathy, or attention i express my thoughts and feelings. this is my outlet where i can say stuff without feeling judged.

After lemmatization

i do not tweet sympathy attention i express my thought feeling my outlet i say stuff without feeling judged

Figure (4.10) Lemmatization Process

4.5 Results of Classification Methods

After applying preprocessing and feature extraction. The data became proper to be classified by machine learning. Three types of machine learning were performed: traditional, deep, and hybrid machine and deep learning.

4.5.1 Results of Applying Traditional Machine Learning Techniques

In this thesis, two dataset were used. Many experiments were performed for each dataset. It included applying four classifiers SVM, LR, DT and LGBM with different preprocessing as stemming, lemmatization, customized removing stop words (means keeping some important stop words and removing others), and replacing slang words& emoji. Four metrics were used to evaluate the performance of classifiers, which are accuracy, precision, recall, and F1-measure, respectively. The experiments were performed on the first original dataset and the original dataset with additional tweets. Tables (4.2), and (4.3) summarize the results of experiments on the first original dataset, and Tables (4.4) and (4.5) show the results of experiments on the first original dataset with additional tweets. Figures (4.11) and (4.12) present the confusion matrix for the best results of all proposed models before and after addition tweets for the first dataset with applying different preprocessing.

Table (4.2) Results of applying TF-IDF with stemming and other different pre-processing on original first dataset.

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.9045	0.9116	0.8910	0.8986	203.925 s	Yes	Yes
LR	0.8862	0.8922	0.8717	0.8792	1.328 s	Customized removing stop words	
LGBM	0.8812	0.8825	0.8697	0.8748	3.307 s	Yes	

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.8812	0.8900	0.8677	0.8749	156.458 s	Yes	Yes
LR	0.8548	0.8616	0.8403	0.8470	1.372 s	Customized removing stop words	
LGBM	0.8522	0,8528	0.8421	0.8462	3.219 s	No	

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.9032	0.9030	0.8895	0.8953	168.067 s	Yes	No
LR	0.8856	0.8863	0.8682	0.8756	1.178 s	Customized removing stop words	
LGBM	0.8730	0.8656	0.8636	0.8646	3.152 s	Yes	

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.8875	0.8952	0.8743	0.8814	157.487 s	Yes	No
LR	0.8642	0.8731	0.8485	0.8562	1.103 s	Customized removing stop words	
LGBM	0.8611	0.8624	0.8502	0.8548	2.864 s	No	

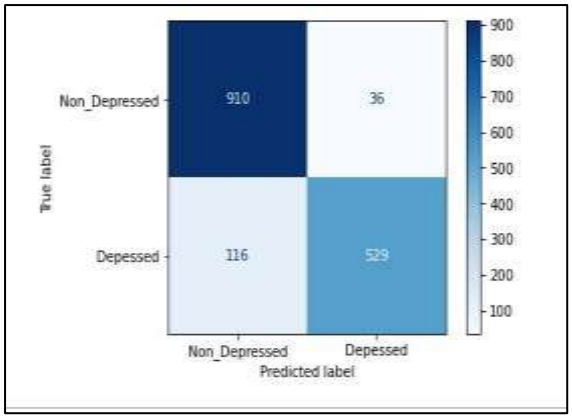
Table (4.3) Results of applying TF_IDF with lemmatization and other different pre-processing on original first dataset.

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.9032	0.9098	0.8901	0.8974	213.253 s	Yes	Yes
LR	0.8874	0.8940	0.8729	0.8805	1.698 s	Customized removing stop words	
LGBM	0.8730	0.8703	0.8652	0.8675	3.462 s	Yes	

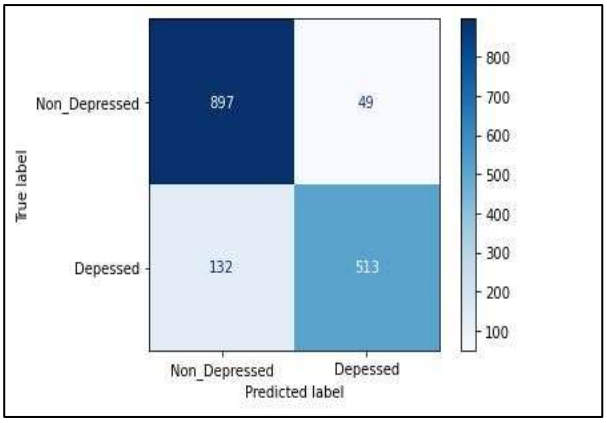
Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.8749	0.8817	0.8533	0.8635	247.793 s	Yes	Yes
LR	0.8567	0.8637	0.8321	0.8428	1.616 s	Customized removing stop words	
LGBM	0.8347	0.8286	0.8191	0.8231	4.296 s	No	

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.8982	0.9025	0.8833	0.8909	229.439 s	Yes	No
LR	0.8787	0.8813	0.8628	0.8699	1.483 s	Customized removing stop words	
LGBM	0.8661	0.8612	0.8567	0.8588	3.924 s	yes	

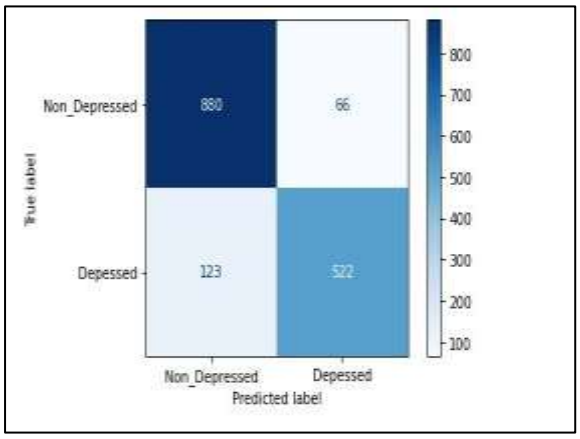
Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.8831	0.8913	0.8672	0.8755	303.302 s	Yes	No
LR	0.8605	0.8671	0.8432	0.8512	2.305 s	Customized removing stop words	
LGBM	0.8448	0.8419	0.8341	0.8374	4.181 s	No	



SVM



LR



LGBM

Figure (4.11) : The confusion matrix with: Customized removing stop words, stemming, Replacing slang & emoji for the SVM, LR, and LGBM algorithms on original first dataset.

Table (4.4) Results of applying TF_IDF with stemming and other different pre-processing after addition tweets.

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.9303	0.9293	0.9316	0.93	296.027 s	Yes	Yes
LR	0.9147	0.9137	0.9151	0.9143	3.371 s	Customized removing stop words	
LGBM	0.9103	0.9093	0.9108	0.9099	4.981 s	Yes	

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.9055	0.9056	0.9074	0.9054	298.654 s	Yes	Yes
LR	0.8952	0.895	0.8967	0.8951	3.217 s	Customized removing stop words	
LGBM	0.8865	0.886	0.8876	0.8863	4.895 s	No	

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.9157	0.9155	0.9172	0.9156	291.304 s	Yes	No
LR	0.9025	0.902	0.9033	0.9023	2.122 s	Customized removing stop words	
LGBM	0.8977	0.897	0.8979	0.8974	4.966 s	Yes	

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.9176	0.9162	0.9186	0.9171	309.201 s	Yes	No
LR	0.9011	0.8996	0.9018	0.9004	2.162 s	Customized removing stop words	
LGBM	0.8884	0.8869	0.8892	0.8877	4.678 s	No	

Table (4.5) Results of applying TF_IDF with lemmatization and other different preprocessing after addition tweets.

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.9196	0.9167	0.9219	0.9186	375.767 s	Yes	Yes
LR	0.9006	0.8976	0.9014	0.8992	3.29 s	Customized removing stop words	
LGBM	0.8933	0.8903	0.8937	0.8917	7.075 s	Yes	

Classifier	Accuracy	Precision	Recall	F1-measure	Time	lemma	Replacing slang & emoji
SVM	0.9147	0.914	0.9154	0.9145	387.283 s	Yes	Yes
LR	0.8982	0.8974	0.8984	0.8978	2.573 s	Customized removing stop words	
LGBM	0.8928	0.892	0.893	0.8924	5.689 s	No	

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.9108	0.9096	0.9123	0.9104	383.177 s	Yes	No
LR	0.9001	0.8989	0.9002	0.8995	3.676 s	Customized removing stop words	
LGBM	0.8821	0.8808	0.8827	0.8815	6.058 s	yes	

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.8977	0.8978	0.9004	0.8975	430.379 s	Yes	No
LR	0.884	0.8832	0.8855	0.8837	2.502 s	Customized removing stop words	
LGBM	0.8689	0.869	0.8714	0.8687	5.686 s	No	

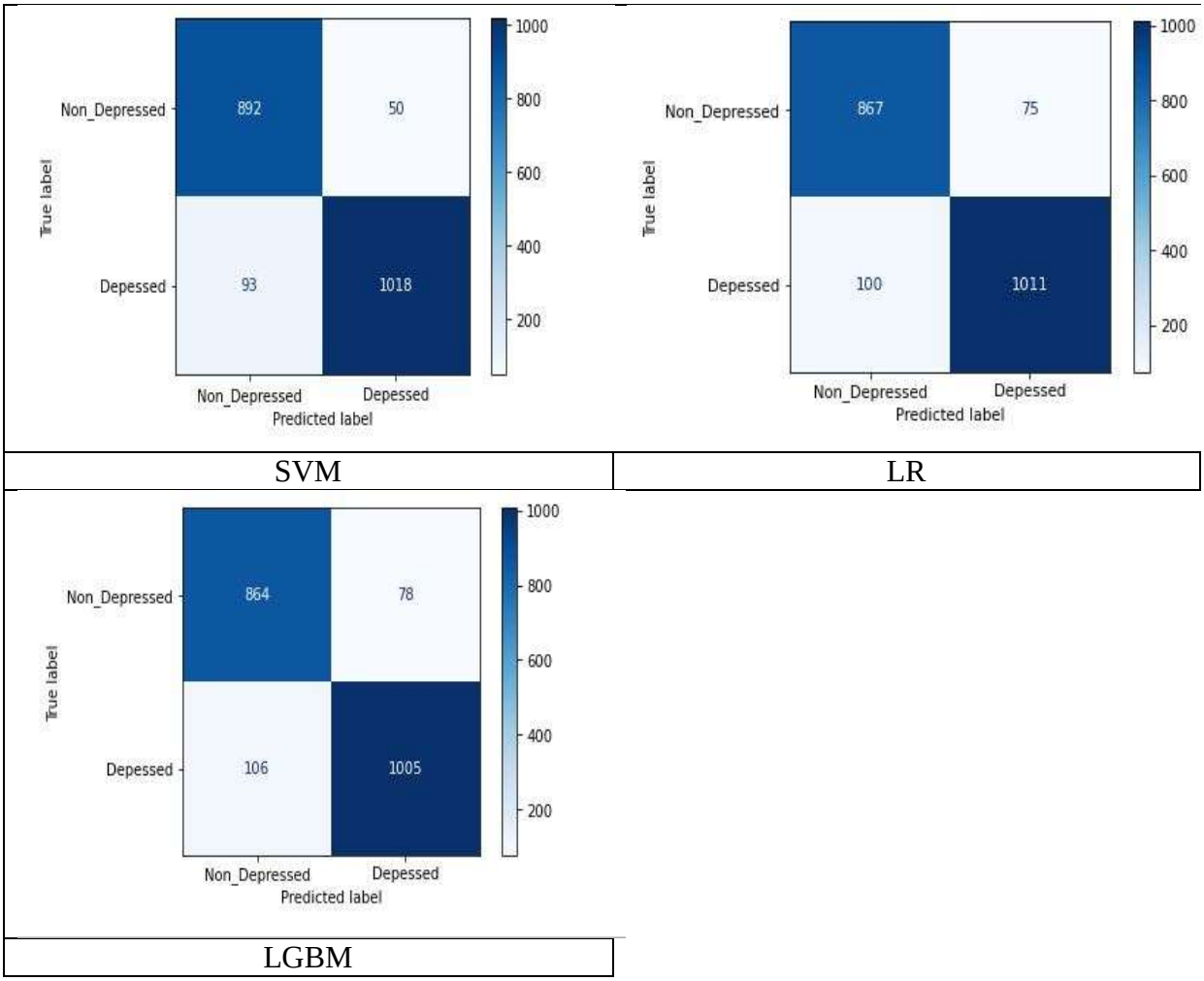


Figure (4.12): The confusion matrix with: Customized removing stop words, stemming, Replacing slang & emoji for the SVM, LR, and LGBM algorithms on original first dataset after addition tweets .

Based on these results, two types of comparisons were performed on the first dataset and second dataset as shown below:

A. Comparison between the Performance of Machine Learning Techniques with the First Dataset

The results in Tables (4.2, 4.3, 4.4, and 4.5) showed that SVM was better than techniques for all metrics and in all experiments. In Table (4.4) can see that SVM achieved the best accuracy, precision, recall, and F1-measure as 0.9303, 0.9293, 0.9316, and 0.93, respectively. On the other hand, the LGBM was the worst in all experiments. The SVM was the best because it worked well with binary problems, and the dataset contained two classes [14]. Therefore, the

results was good. However, the findings showed that SVM takes longer time than others classifiers. While LR was faster than others, but its performance was not better.

In addition, the results in Tables (4.4) and (4.5) show that the performance for all techniques was enhanced after adding more tweets. As well as, the execution time becomes longer than before adding tweets. Execution time increases with more data.

B. Comparison between Different Pre-processing on the First Dataset

The findings in Tables (4.2, 4.3, 4.4, and 4.5) suggested that application stemming, removing some stop words, and replacing slang words & emojis can lead to the best results for all metrics and with all classifiers. Otherwise, the results were the worst without using preprocessing. In addition, the results showed that all metrics increased when using customized removing stop words and decreased when deleting all. The best results was with SVM. There were many reasons for these results; the first reason was that stemming reduced the derived words to their root word. So, it reduced the number of words in the text and made it in one form. The second was not removing any important stop words such as pronouns, negation words, and other words. These words are used by depressed people frequently, where pronouns express themselves. Also, removing negation words changed the meaning of the sentence from a sad sentence to a natural sentence. In the end, deleting all these affected on the performance of classifiers. Third, replacing slang words with standard terms to make all words in one form. Forth, replacing emoji with corresponding words instead of removing them. The emoji expresses a person's feelings such as sadness or happiness and removing it may change the meaning of the sentence. Finally, adding a number of depressed tweets led to enhance the accuracy and

other metrics. In addition, lemmatization needed time more than stemming as can see in the results.

C. Comparison between The Performance of Machine Learning Techniques on the Second Dataset

The outcomes of applying machine learning on the second dataset with a different pre-processing show that LGBM outperformed other techniques and with all metrics see (Tables 4.6 and 4.7). Figure (4.13) shows the confusion matrix for LGBM.

Table (4.6) Results of applying TF_IDF with stemming and other different pre-processing with second dataset.

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.9954	0.9955	0.9954	0.9954	376.407 s	Yes	Yes
LR	0.9939	0.9939	0.9939	0.9939	0.307 s	Customized removing stop words	
LGBM	0.9971	0.9970	0.9971	0.9970	10.98 s	Yes	
DT	0.9400	0.9433	0.9416	0.9400	17.129 s		
Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.9971	0.9971	0.9970	0.9970	5710.547 s	Yes	Yes
LR	0.9963	0.9963	0.9962	0.9962	6.2 s	Customized removing stop words	
LGBM	0.9972	0.9972	0.9972	0.9972	35.211 s	No	
DT	0.9364	0.9402	0.9384	0.9364	15.36s		
Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.9962	0.9962	0.9961	0.9962	12232.42 s	Yes	No
LR	0.9947	0.9946	0.9946	0.9946	7.619 s	Customized removing stop words	
LGBM	0.9966	0.9965	0.9966	0.9966	29.365 s	Yes	
DT	0.9376	0.9411	0.9393	0.9375	23.22 s		
Classifier	Accuracy	Precision	Recall	F1-measure	Time	Stemming	Replacing slang & emoji
SVM	0.9966	0.9967	0.9966	0.9966	5300.805 s	Yes	No
LR	0.9957	0.9957	0.9957	0.9957	8.278 s	Customized removing stop words	
LGBM	0.9976	0.9976	0.9976	0.9976	27.421 s	No	
DT	0.9400	0.9427	0.9421	0.9400	15.272 s		

The best Accuracy, Precision, recall, and F1-measure for LGBM was 0.9980 for all metrics as can see in Table (4.7). Three main reasons may be led to the superiority of LGBM, the first reason it uses a leaf-wise split strategy rather than a level-wise split approach which produces much more complex trees. Second, LGBM can perform as well with large datasets. Third, LGBM

merges more than one weak model into one model. While the DT was the worst technique in all experiments this may be because DT tend to overfit. In addition, any change in data can lead to change in results. Moreover, DT works well with a small dataset. On other hand, SVM and LR achieved results close to LGBM between 0.9947 and 0.9972. When compare between execution times for each method can see that SVM has the longest time between them while LR was the shortest time.

Table (4.7) Results of applying TF_IDF with lemmatization and other different preprocessing on second dataset.

Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.9969	0.9969	0.9969	0.9969	6466.788 s	Yes	Yes
LR	0.9959	0.9959	0.9959	0.9959	7.536 s	Customized removing stop words	
LGBM	0.9980	0.9980	0.9980	0.9980	35.83 s	Yes	
DT	0.8826	0.8996	0.8868	0.8820	16.253 s		
Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.9969	0.9969	0.9969	0.9969	6400.369 s	Yes	Yes
LR	0.9952	0.9952	0.9952	0.9952	8.369 s	Customized removing stop words	
LGBM	0.9978	0.9978	0.9978	0.9978	32.094 s	No	
DT	0.8796	0.8977	0.8833	0.8788	16.242s		
Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.9972	0.9972	0.9971	0.9971	5516.184 s	Yes	No
LR	0.9953	0.9953	0.9952	0.9953	6.878 s	Customized removing stop words	
LGBM	0.9979	0.9978	0.9979	0.9979	29.937 s	yes	
DT	0.8746	0.8940	0.8793	0.8739	15.879 s		
Classifier	Accuracy	Precision	Recall	F1-measure	Time	Lemma	Replacing slang & emoji
SVM	0.9963	0.9963	0.9962	0.9962	5676.929 s	Yes	No
LR	0.9952	0.9952	0.9952	0.9952	13.864 s	Customized removing stop words	
LGBM	0.9979	0.9978	0.9979	0.9979	29.937 s	No	
DT	0.8781	0.8964	0.8815	0.8773	16.722s		

D. Comparison between Different Pre-processing on the Second Dataset

Tables (4.6) and (4.7) explains the results of applying different pre-processing with traditional machine learning techniques. When noticing the results of applying stemming and customized removing stop words in Table (4.6), can see that performance of SVM, LR, and LGBM reduces when using

both preprocessing and increases when applying stemming only. On the other hand, DT is best when preformed stemming and customized removing stop words or using stemming only. While DT is worst when not applying customized removing stop words.

In general, the performance of SVM, LR, and LGBM is better when applying lemmatization than stemming as can see in Table (4.7), especially when performed with customized removing stop words, conversely, DT is not like them.

There is a difference in the results between the algorithms, as some models increase their performance and others decrease with the use of the proposed preprocessing. In addition, the outcomes are very close to each other, except for the DT. The reasons for this difference may be due to the large size of the dataset, and thus the increase in the size of the training data then increases the accuracy. Another reason why the results are not affected by the proposed preprocessing may be that the data does not contain emoji and slang words.

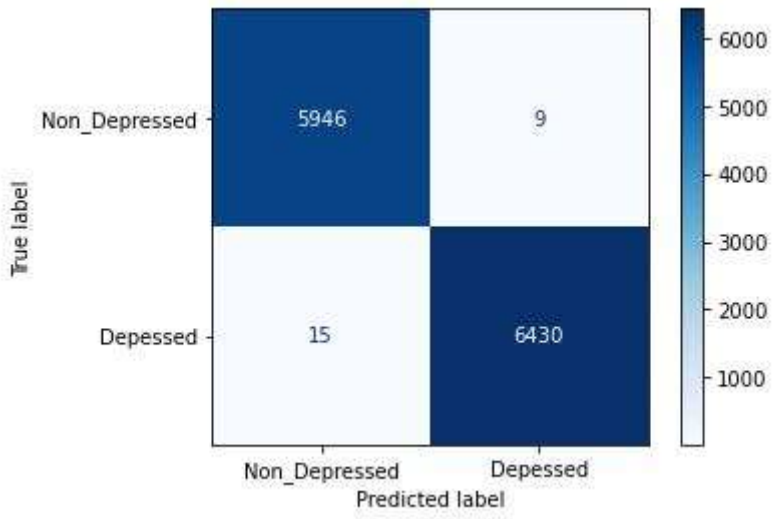


Figure (4.13): The confusion matrix with: Customized removing stop words, lemmatization, replacing slang & emoji for the LGBM algorithm on second dataset.

4.5.2 Results of Applying Deep Learning Techniques

In this thesis, two deep learning techniques were used CNN and Bi-LSTM. The features extraction method applied is word2vec. The preprocessing that used is cleaning only without stemming or removing stop words to get the best accuracy. Table (4.8) shows the outcomes of performing CNN and Bi-LSTM. Figures (4.14) and (4.15) explains the confusion matrix for CNN and Bi-LSTM.

Table (4.8) the results of applying CNN and Bi-LSTM on the first and second datasets.

	Classifier	Accuracy	Precision	Recall	F1-measure
First dataset	Bi-LSTM	0.9454	0.9497	0.9405	0.9441
	CNN	0.9323	0.9307	0.9347	0.9319
Second dataset	Bi-LSTM	0.9961	0.9961	0.9960	0.9961
	CNN	0.9971	0.9971	0.9971	0.9971

The results show that Bi-LSTM outperforms CNN in the first dataset with 94% for all metrics. There are many reasons for this, Bi-LSTM is able to process sequence data such as text because it has the memory to remember previous and future words [80]. In addition, Bi-LSTM is common for its ability to retain the chronological order between data, and this is very important when analyzing lengthy textual opinions [81]. Also, Bi-LSTM's strong ability to extract advanced text information plays a significant role in text classification [63]. While the performance of CNN model was less than that of Bi-LSTM model. The cause may be that CNN can extract local features from various locations in the text but ignores the contextual features for words [29] in addition to the strong features of Bi-LSTM mentioned above. The performance of LSTM is better than that of CNN, but it needs more time and is more accurate with long sentences [17]. While the second dataset shows that CNN exceeds Bi-LSTM with 99.71 for all metrics. The outcomes are too close to each other, the reason for the high performance of the two models may be due to their training on a lot of data.

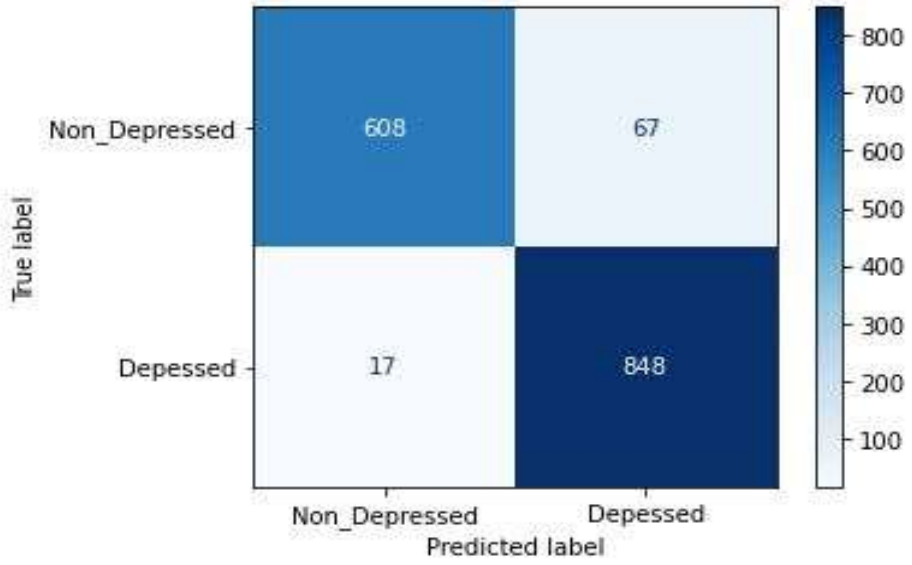


Figure (4.14) Confusion matrix Bi-LSTM with first dataset.

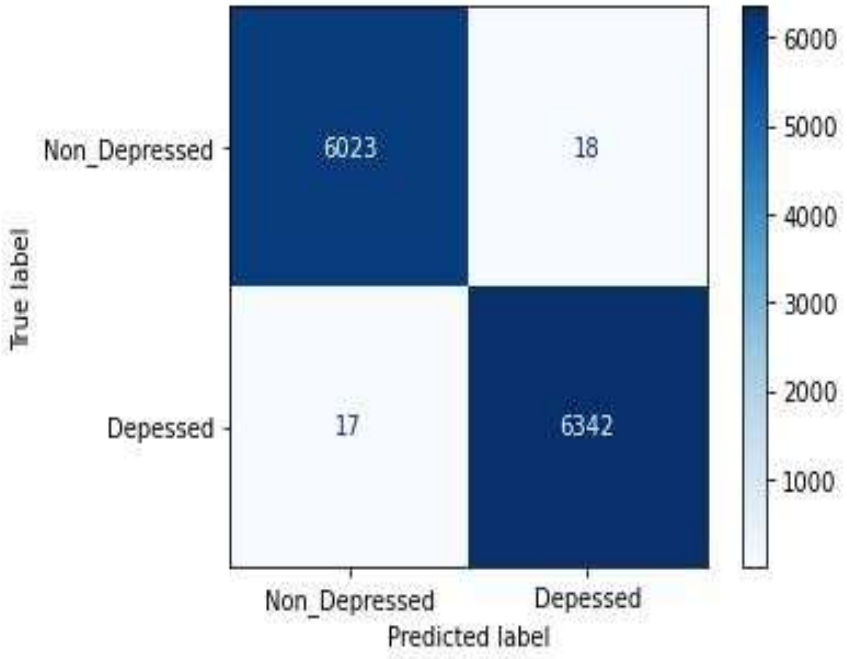


Figure (4.15) Confusion matrix for CNN with second dataset

When seeing Figures (4.16) and (4.17) to show the performance of the training stage for Bi-LSTM with the first dataset we notice that the training accuracy naturally increases during the epochs and the training loss decreases as well. While the accuracy of validation in the first three epochs increases naturally and loss validation also decreases, in the fourth epoch the loss

increases dramatically and the accuracy decreases, and this causes overfitting and can prevent by early stopping, as well as the use of dropouts and generalization.

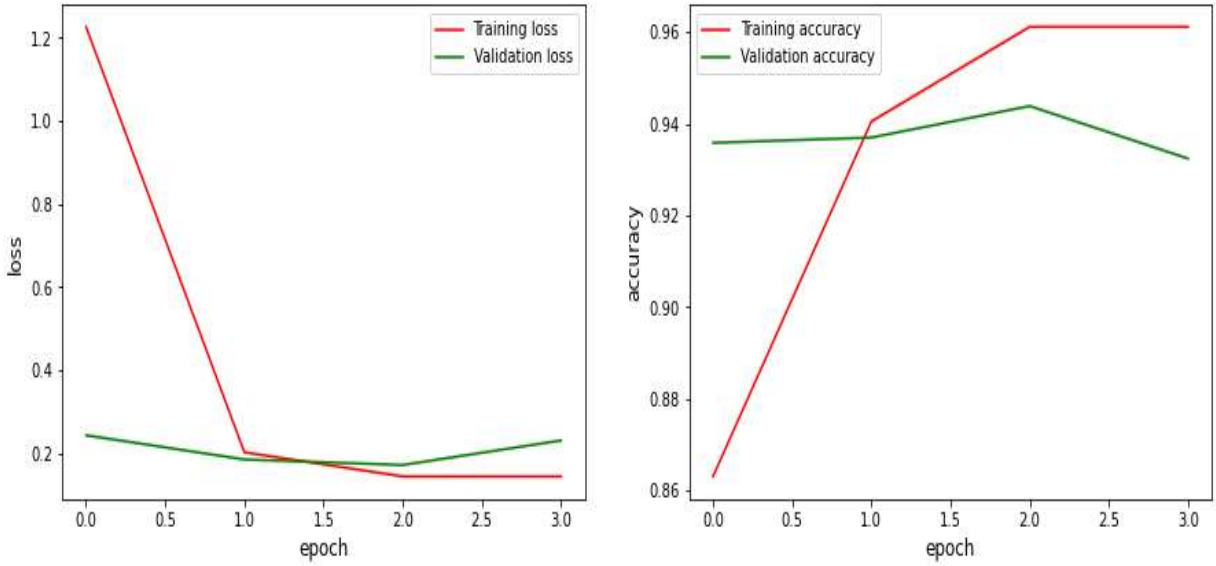


Figure (4.16) Loss and accuracy for Bi-LSTM with first dataset

```
Epoch 1/4
123/123 [==============================] - 1319s 11s/step - loss: 1.2261 - accuracy: 0.8631 - val_loss: 0.2441 - val_accuracy:
0.9359
Epoch 2/4
123/123 [==============================] - 2540s 21s/step - loss: 0.2032 - accuracy: 0.9405 - val_loss: 0.1860 - val_accuracy:
0.9370
Epoch 3/4
123/123 [==============================] - 1233s 10s/step - loss: 0.1452 - accuracy: 0.9612 - val_loss: 0.1727 - val_accuracy:
0.9439
Epoch 4/4
123/123 [==============================] - 1236s 10s/step - loss: 0.1452 - accuracy: 0.9612 - val_loss: 0.2317 - val_accuracy:
0.9324
```

Figure (4.17) Training stage for Bi-LSTM with first dataset that show loss and accuracy.

While the second dataset with no problem with overfitting (see Figure 4.18 and 4.19).

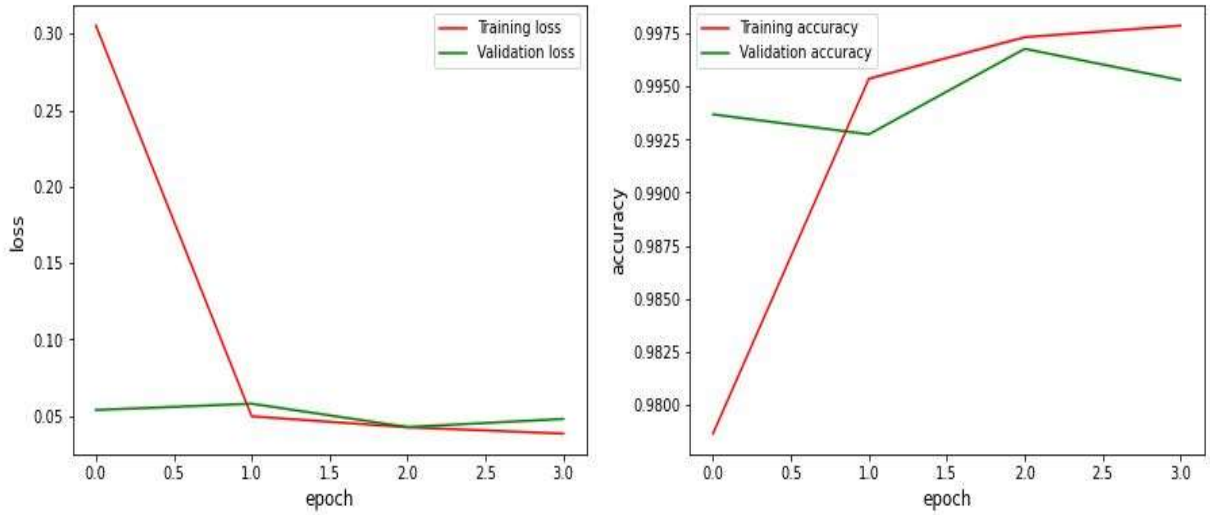


Figure (4.18) Loss and accuracy for Bi-LSTM with second dataset

```
history=model_w2vec.fit(x_train, y_train, batch_size=64,validation_data=(x_val, y_val), epochs=4)

Epoch 1/4
543/543 [==============================] - 4547s 8s/step - loss: 0.3053 - accuracy: 0.9786 - val_loss: 0.0539 - val_accuracy:
0.9937
Epoch 2/4
543/543 [==============================] - 5268s 10s/step - loss: 0.0498 - accuracy: 0.9954 - val_loss: 0.0581 - val_accuracy:
0.9927
Epoch 3/4
543/543 [==============================] - 4671s 9s/step - loss: 0.0426 - accuracy: 0.9973 - val_loss: 0.0428 - val_accuracy:
0.9968
Epoch 4/4
543/543 [==============================] - 4642s 9s/step - loss: 0.0385 - accuracy: 0.9979 - val_loss: 0.0481 - val_accuracy:
0.9953
```

Figure (4.19) Training stage for Bi-LSTM with second dataset that show loss and accuracy.

4.5.3 Results of Applying Hybrid Deep Learning with Machine Techniques

This thesis conducted various experiments combining deep learning and machine learning techniques to take advantage of each technique where deep learning method used to extract features and train model while the machine learning technique applied to classify tweets only.

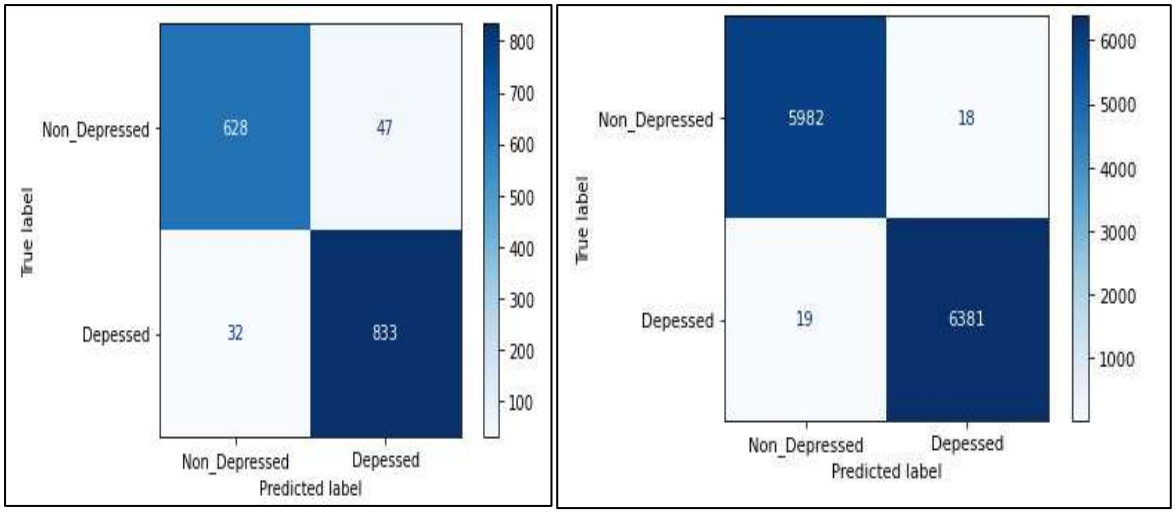
It suggested Bi-LSTM and CNN as deep learning and SVM, LGBM, and XGBOOST as machine learning. The results of the combination in Table 4.9 are shown that the hybrid model Bi-LSTM-XGBOOST produce the best accuracy,

precision, recall, and F1-measure as 0.9487, 0.9490, 0.9466, and 0.9477, respectively with the first dataset. In addition, Bi-LSTM-XGBOOST obtain the best outcomes 0.9970 for all metrics when applying on the second dataset.

Table (4.9) the results of applying hybrid deep and machine learning on the first and second datasets.

	Classifier	Accuracy	Precision	Recall	F1-measure
First dataset	Bi-LSTM-SVM	0.9428	0.9422	0.9416	0.9419
	Bi-LSTM-LGBM	0.9435	0.9427	0.9425	0.9426
	Bi-LSTM-XGBOOST	0.9487	0.9490	0.9466	0.9477
	CNN-SVM	0.9392	0.9378	0.9398	0.9387
	CNN-LGBM	0.9380	0.9366	0.9383	0.9374
	CNN-XGBOOST	0.9380	0.9367	0.9382	0.9374
Second dataset	Bi-LSTM-SVM	0.9962	0.9962	0.9961	0.9962
	Bi-LSTM-LGBM	0.9965	0.9965	0.9965	0.9965
	Bi-LSTM-XGBOOST	0.9970	0.9970	0.9970	0.9970
	CNN-SVM	0.9968	0.9969	0.9968	0.9968
	CNN-LGBM	0.9968	0.9968	0.9968	0.9968
	CNN-XGBOOST	0.9962	0.9962	0.9961	0.9962

Bi-LSTM-XGBOOST superiority over the rest of the models may be due to the power of the mix of Bi-LSTM Which is distinguished by its ability to remember previous and future words in two directions and prepare data to train XGBOOST and then classify it by XGBOOST. In addition, XGBOOST performs more accurately and efficiently. To fix the mistakes made by earlier models, it combines multiple weak models into a single model. Figure 4.20 shows the confusion matrix for Bi-LSTM-XGBOOST model with the two dataset.



Bi-LSTM-XGBOOST with first dataset Bi-LSTM-XGBOOST with second dataset

Figure (4.20) Confusion matrix for Bi-LSTM-XGBOOST with first and second dataset

4.5.4 Results of applying 10-fold on the original first dataset

Table (4.10) shows the results of applying 10-fold on the first dataset. The results indicated that the 10-fold method produced outcomes close to those of the conventional way of splitting, hence the traditional method was employed for data splitting.

Table (4.10) The results of applying 10-fold with original first dataset.

Classifier	Accuracy	Precision	Recall	F1-measure	Stemming	Replacing slang & emoji
SVM	0.9070	0.9336	0.8324	0.8799	Yes	Yes
LR	0.8815	0.9023	0.7971	0.8463	Customized removing stop words	
LGBM	0.8891	0.8805	0.8439	0.8618	Yes	
XGBOOST	0.8909	0.8789	0.8511	0.8647		

4.5.5 Comparison between the Results of Different Methods

When compare between the results of different proposed models that applied on the first dataset and the second dataset many notices can see (see Tables 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9).

With the first dataset can see that the hybrid model Bi-LSTM-XGBOOST (see Table 4.9) outperform others suggested models. The primary causes for that may be due to first, the power of combination between Bi-LSTM and

XGBOOST, where Bi-LSTM is characterized by its ability to process texts and remember the previous and future words, so, can extracts important features from text that used to train XGBOOST in the next classification step. Second, XGBOOST is one of the most common machine learning techniques that used nowadays. It combines between many weak models into single model and this lead to more accuracy. Third, extracting features from text using word2vec is help also because in word2vec, the context of the sentence and its meaning are important, while TF-IDF ignores the order of words in a sentence and focuses on the frequency of words. However, the first data set was suffered from overfitting and this led to reduce the accuracy.

On other hand, the outcomes of applying various models on the second dataset clarify that the results of classifiers are too close to each other. However, the LGBM is the best for all metrics (see Table 4.7). Many reasons may illustrate that one of these is the power of LGBM in classification because it integrates many weak models in one model. Furthermore, the second dataset is big data so there are many samples to train models.

4.6 Comparison with Other Related Works

This thesis aims to classify the tweets as depressed or not by comparing the performance of three types of machine learning techniques (traditional machine learning, deep learning, hybrid deep learning, and traditional machine learning). Table (4.11) shows the comparison between the outcomes of this thesis and the results of other related works on the first dataset before adding tweets.

From Table (4.11) can notice that the suggested traditional machine learning SVM with TF-IDF in this thesis outperformed the study [25]. The preprocessing suggested in this thesis play an important role in improving performance. It includes three preprocessing stemming, customized removing stop words, and replacing slang & emoji. The last two preprocessing

customized removing stop words, and replacing slang & emoji were not applied in the study [25] and other previous studies.

In addition, CNN with word2vec exceed the models in the study [28]. This thesis applied cleaning processing that mentioned in Chapter Three only without stemming or customized removing stop words preprocessing to get the best performance for all metrics. The reason for this may be word2vec trained on millions of words with the adjacent words, when deleting a word or returning it to its origin, as in stemming, it may lead to a decrease in performance.

Table (4.11) the comparison between the outcomes of this thesis and the results of other related works on the first dataset before adding tweets.

study	Classifier	Mean accuracy	Precision	Recall	F1-measure	Stemming	Replacing slang & emoji
[25]	SVM+TF-IDF	85	-	-	-	Yes	No
						Customized removing stop words	
						No	
						Lemmatization	
						Yes	
	Classifier	Accuracy	Precision	Recall	F1-measure	Stemming	Replacing slang & emoji
[28]	**CNN+word2vec**	**84.8**	**0.85**	**0.85**	**0.84**	Yes	No
						Customized removing stop words	
						No	
						Lemmatization	
						Yes	
	Classifier	Accuracy	Precision	Recall	F1-measure	Stemming	Replacing slang & emoji
Proposed system	SVM+TF-IDF	0.9045	0.9116	0.8910	0.8986	Yes	Yes
						Customized removing stop words	
						Yes	
						Lemmatization	
						No	
	Classifier	Accuracy	Precision	Recall	F1-measure	stemming	Replacing slang & emoji
Proposed system	**CNN+word2vec**	**0.9076**	**0.9030**	**0.9078**	**0.9051**	No	yes
						Customized removing stop words	
						No	
						Lemmatization	
						No	

Chapter Five

Conclusion and Future Works

CHAPTER FIVE

CONCLUSION AND FUTURE WORKS

5.1. Conclusions

Nowadays, depression is like as a silent killer, which makes people kill themselves. Therefore, it is necessary to fight it in all possible ways. Thus, this thesis aims to create an efficient model for the classification of depression among users of Twitter. Numerous experiments were conducted in this thesis using various preprocessing, feature extraction approaches, and machine learning algorithms then comparing them to determine which was the most effective.

Based on the thesis outcomes, many conclusions can be drawn:

1) Most techniques get close results to each other especially, with the second dataset. However, the outcomes indicate that LGBM led to the best results with 99% as traditional machine learning with the second dataset. Mixing deep learning architecture with traditional machine learning techniques produces the greatest results, where Bi-LSTM-XGBOOST with 94% as the best with the first dataset.

2) According to the findings, there are various factors contributed in improving performance:

A. Recommended preprocessing significantly enhances the performance of traditional techniques, where stemming, customized removing stop words, and replacing slang & emoji help in improving the performance of all classifiers with the first dataset and TF-IDF feature extraction method.

B. The size of the data is the most common factor in performance improvement, as the thesis found that accuracy improves as data size increases.

5.2. Future Works

Despite having high performance, every model has its drawbacks. In future we need to :

1) Applied models on other social media such as Facebook, Instagram, and Reddit.
2) Working with text written in Arabic language.
3) Using other types of data such as images, video, etc.
4) Preforming others techniques.

References

[1] G. C. Adeyanju et al., “Behavioural symptoms of mental health disorder such as depression among young people using Instagram: a systematic review,” *Translational Medicine Communications*, Vol. 6, No. 1, 2021, doi: 10.1186/s41231-021-00092-3 .

[2] O. E. Daly, “Physical illness in those with mental illness: Psychiatric services need to change focus,” *British Journal of Psychiatry*, Vol. 210, No. 1, 2017, doi: 10.1192/bjp.bp.116.182915.

[3] H. Owadh, R. A. Ghaleb, and F. Adnan, “Forty Micromole Hydroxychloroquine Enhanced Cytotoxic Effect of Forty Micromole Hydroxychloroquine Enhanced Cytotoxic Effect of Doxorubicin Against Laryngeal Cancer Cell Line HEp-2,” *Journal of Research in Pharmacy* , Vol. 26, No. 4,pp. 714-721, 2022, doi: 10.29228/jrp.169.

[4] WHO, “Depression,” September 13 2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/depression.

[5] E. M. Seabrook, M. L. Kern, B. D. Fulcher, and N. S. Rickard, “Predicting depression from language-based emotion dynamics: Longitudinal analysis of facebook and twitter status updates,” *Journal of Medical Internet Research*, Vol. 20, No. 5, 2018, doi: 10.2196/jmir.9267.

[6] J. C. Kaufman and J. D. Sexton, “Why doesn’t the writing cure help poets?,” *Review of General Psychology*, Vol. 10, No. 3, pp. 268–282, 2006, doi: 10.1037/1089-2680.10.3.268.

[7] K. Searle, G. Blashki, R. Kakuma, H. Yang, Y. Zhao, and H. Minas, “Current needs for the improved management of depressive disorder in community healthcare centres, shenzhen, china: A view from primary care medical leaders,” *International Journal of Mental Health Systems*, Vol. 13, No. 47, 2019, doi: 10.1186/s13033-019-0300-0.

[8] K. Sudha, S. Sreemathi, B. Nathiya, and D. RahiniPriya, “Depression Detection using Machine Learning,” In: *International Journal of Research and Advanced Development*. *on AICTE Sponsored International Conference on Data Science & Big Data Analytics for Sustainability*, 2020.

[9] B. Levis *et al.*, “Patient Health Questionnaire-9 scores do not accurately estimate depression prevalence: individual participant data meta-analysis,” *Journal of Clinical Epidemiology*, Vol. 122, pp. 115-128, 2020, doi: 10.1016/j.jclinepi.2020.02.002.

[10] H. Patidar, J. Umre, “PREDICTING DEPRESSION LEVEL USING SOCIAL MEDIA POSTS,” *International Journal of Research*. Vol. 8, No. 12, pp. 234 – 237, 2020. DOI: https://doi.org/10.29121/granthaalayah.v8.i12.2020.1972

[11] A. Naghavi, T. Teismann, Z. Asgari, M. R. Mohebbian, M. Mansourian, and M. Á. Mañanas, “Accurate diagnosis of suicide ideation/behavior using robust ensemble machine learning: A university student population in the middle east and north africa (mena) region,” *Diagnostics*, Vol. 10, No. 11, 2020, doi: 10.3390/diagnostics10110956.

[12] N. P. Shetty, B. Muniyal, A. Anand, S. Kumar, and S. Prabhu, “Predicting depression using deep learning and ensemble algorithms on raw twitter data,” *International Journal of Electrical and Computer Engineering*, Vol. 10, No. 4, pp. 3751–3756, 2020, doi: 10.11591/ijece.v10i4.pp3751-3756.

[13] A. R. Javed, M. U. Sarwar, M. O. Beg, M. Asim, T. Baker, and H. Tawfik, "A collaborative healthcare framework for shared healthcare plan with ambient intelligence," *Human-centric Computing and Information Sciences*, Vol. 10, No. 40, 2020, doi: 10.1186/s13673-020-00245-7.

[14] E. K. Al-Yasiri and A. Al-Azawei, "Improving Arabic sentiment analysis on social media: A comparative study on applying different pre-processing techniques," *Compusoft*, Vol. 8, No. 6, pp. 3150–3157, 2019.

[15] T. Zhang, A. M. Schoene, S. Ji, and S. Ananiadou, "Natural language processing applied to mental illness detection: a narrative review," *npj Digital Medicine*, Vol. 5, No. 46, 2022, doi: 10.1038/s41746-022-00589-7.

[16] A. Ashraf, T. S. Gunawan, B. S. Riza, E. V. Haryanto, and Z. Janin, "On the review of image and video-based depression detection using machine learning," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 3, pp. 1677–1684, 2020, doi: 10.11591/ijeecs.v19.i3.pp1677-1684.

[17] C. N. Dang, M. N. Moreno-García, and F. De La Prieta, "Hybrid Deep Learning Models for Sentiment Analysis," Complexity, vol. 2021, 2021, doi: 10.1155/2021/9986920.

[18] C. Bhargava, S. Poornima, S. Mahur, and M. Pushpalatha, "Depression Detection Using Sentiment Analysis of Tweets," Turkish Journal of Computer and Mathematics Education, vol. 12, no. 11, pp. 5411–5418, 2021.

[19] Y. Dong, X. Yang, "A hierarchical depression detection model based on vocal and emotional cues," *Neurocomputing* , Vol. 441, No. 1, ,pp. 279–290, 2021 , https://doi.org/10.1016/j.neucom.2021.02.019 .

[20] Y. DIng, X. Chen, Q. Fu, and S. Zhong, "A Depression Recognition Method for College Students Using Deep Integrated Support Vector Algorithm," *IEEE Access*, Vol. 8, pp. 75616–75629, 2020, doi: 10.1109/ACCESS.2020.2987523.

[21] S. Pandya, A. Sur, and N. Solke, "COVIDSAVIOR: A Novel Sensor-Fusion and Deep Learning Based Framework for Virus Outbreaks," *Frontiers in Public Health*, Vol. 9, 2021, doi: 10.3389/fpubh.2021.797808.

[22] F. Azam, M. Agro, M. Sami, M. H. Abro, and A. Dewani, "Identifying Depression among Twitter Users using Sentiment Analysis," 2021 International Conference on Artificial Intelligence, ICAI 2021, no. April, pp. 44–49, 2021, doi: 10.1109/ICAI52203.2021.9445271.

[23] H. Alsagri and M. Ykhlef, "Machine Learning-based Approach for Depression Detection in Twitter Using Content and Activity Features.", *IEICE TRANS. INF. & SYST*, Vol. E103, No. 8, pp. 1825-1832, 2020, https://doi.org/10.1587/transinf.2020EDP7023

[24] Kumar, A. Sharma, A. Arora, "Anxious Depression Prediction in Real-time Social Data", *In: International Conf.on Advanced Engineering, Science, Management and Technology*, Uttaranchal University, Dehradun, India,2019, https://doi.org/10.2139/ssrn.3383359.

[25] A. Amanat *et al.*, "Deep Learning for Depression Detection from Textual Data," *Electronics (Switzerland)*, Vol. 11, No. 676, 2022, doi: 10.3390/electronics11050676.

[26] H.K. Cho, "Twitter Depression Dataset," Kaggle. 2021 [Online]. Available: https://www.kaggle.com/datasets/hyunkic/twitter-depression-dataset .

[27] M. Z. Uddin, K. K. Dysthe, A. Følstad, and P. B. Brandtzaeg, "Deep learning for prediction of depressive symptoms in a large textual dataset," Neural Computing and Applications, vol. 34, no. 1, pp. 721–744, 2022, doi: 10.1007/s00521-021-06426-4.

[28] V. Tejaswini, K. S. Babu, and B. Sahoo, "Depression Detection from Social Media Text Analysis using Natural Language Processing Techniques and Hybrid Deep Learning Model," ACM Transactions on Asian and Low-Resource Language Information Processing, 2022, doi: 10.1145/3569580.

[29] H. Kour and M. K. Gupta, An hybrid deep learning approach for depression prediction from user tweets using feature-rich CNN and bi-directional LSTM, vol. 81, no. 17. Multimedia Tools and Applications, 2022. doi: 10.1007/s11042-022-12648-y.

[30] T. F. Mumu, I. J. Munni, and A. K. Das, "Depressed People Detection from Bangla Social Media Status using LSTM and CNN Approach," Journal of Engineering Advancements, vol. 2, no. 01, pp. 41–47, 2021, doi: 10.38032/jea.2021.01.006.

[31] M. M. Aldarwish and H. F. Ahmad, "Predicting Depression Levels Using Social Media Posts," *Proceedings - 2017 IEEE 13th International Symposium on Autonomous Decentralized Systems, ISADS 2017*, vol. 1, pp. 277–280, 2017, doi: 10.1109/ISADS.2017.41.

[32] R. Martínez-Castaño, J. C. Pichel, and D. E. Losada, "A big data platform for real time analysis of signs of depression in social media," *International Journal of Environmental Research and Public Health*, vol. 17, no. 13, pp. 1–23, 2020, doi: 10.3390/ijerph17134752.

[33] R. Safa, P. Bayat, and L. Moghtader, *Automatic detection of depression symptoms in twitter using multimodal analysis*, vol. 78, no. 4. Springer US, 2022. doi: 10.1007/s11227-021-04040-8.

[34] L. Lin, X. Chen, Y. Shen, and L. Zhang, "Towards automatic depression detection: A bilstm/1d cnn-based model," *Applied Sciences (Switzerland)*, vol. 10, no. 23, pp. 1–20, 2020, doi: 10.3390/app10238701.

[35] A. H. Wang, "Detecting spam bots in online social networking sites: A machine learning approach," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6166 LNCS, pp. 335–342, 2010, doi: 10.1007/978-3-642-13739-6_25.

[36] L. Mariñelarena-dondena, M. Sapino, and M. L. Errecalde, "ORIGINALES / ORIGINAL PAPERS de distintos enfoques de aprendizaje automático comparativo de diferentes enfoques da," pp. 42–54, 2017.

[37] M. De Choudhury, S. Counts, and E. Horvitz, "Social media as a measurement tool of depression in populations," *Proceedings of the 5th Annual ACM Web Science Conference, WebSci'13*, vol. volume, pp. 47–56, 2013, doi: 10.1145/2464464.2464480.

[38] K. Loveys, P. Crutchley, E. Wyatt, and G. Coppersmith, "Small but Mighty: Affective Micropatterns for Quantifying Mental Health from Social Media Language," no. March, pp. 85–95, 2017, doi: 10.18653/v1/w17-3110.

[39] A. Leis, F. Ronzano, M. A. Mayer, L. I. Furlong, and F. Sanz, "Detecting signs of depression in tweets in Spanish: Behavioral and linguistic analysis," *Journal of Medical Internet Research*, vol. 21, no. 6, 2019, doi: 10.2196/14199.

[40] E. K. M. Al-yasiri, "Arabic Sentiment Analysis for Identifying Terrorism Supporters on Twitter Using Data Mining Techniques," 2019.

[41] K. P.Y., R. Dube, S. Barbade, G. Kulkarni, N. Konda, and M. Konkati, "Depression Detection using Machine Learning," *SSRN Electronic Journal*, pp. 1–6, 2021, doi: 10.2139/ssrn.3851975.

[42] D. M. Boyd and N. B. Ellison, "Social network sites: Definition, history, and scholarship," *Journal of Computer-Mediated Communication*, vol. 13, no. 1, pp. 210–230, 2007, doi: 10.1111/j.1083-6101.2007.00393.x.

[43] S. S. Dash, S. K. Nayak, and D. Mishra, "A review on machine learning algorithms," *Smart Innovation, Systems and Technologies*, vol. 153, no. October, pp. 495–507, 2021, doi: 10.1007/978-981-15-6202-0_51.

[44] M. M. Truşcă, "Efficiency of SVM classifier with Word2Vec and Doc2Vec models," *Proceedings of the International Conference on Applied Statistics*, vol. 1, no. 1, pp. 496–503, 2019, doi: 10.2478/icas-2019-0043.

[45] ChatGPT, 'SVM pseudocode', https://chat.openai.com/chat , (accessed March 9, 2023).

[46] F. Alzamzami, M. Hoda, and A. El Saddik, "Light Gradient Boosting Machine for General Sentiment Classification on Short Texts: A Comparative Evaluation," IEEE Access, vol. 8, pp. 101840–101858, 2020, doi: 10.1109/ACCESS.2020.2997330 .

[47] M. Al-kasassbeh, M. A. Abbadi, and A. M. Al-Bustanji, "LightGBM Algorithm for Malware Detection," Advances in Intelligent Systems and Computing, vol. 1230 AISC, no. July, pp. 391–403, 2020, doi: 10.1007/978-3-030-52243-8_28.

[48] A. Bin Asad, R. Mansur, S. Zawad, N. Evan, and M. I. Hossain, "Analysis of Malware Prediction Based on Infection Rate Using Machine Learning Techniques," 2020 IEEE Region 10 Symposium, TENSYMP 2020, no. June, pp. 706–709, 2020, doi: 10.1109/TENSYMP50017.2020.9230624.

[49] M. Liang, Z. Chang, Z. Wan, Y. Gan, E. Schlangen, and B. Šavija, "Interpretable Ensemble-Machine-Learning models for predicting creep behavior of concrete," Cement and Concrete Composites, vol. 125, no. September 2021, 2022, doi: 10.1016/j.cemconcomp.2021.104295.

[50] A. A. Taha and S. J. Malebary, "An Intelligent Approach to Credit Card Fraud Detection Using an Optimized Light Gradient Boosting Machine," *IEEE Access*, vol. 8, no. February, pp. 25579–25587, 2020, doi: 10.1109/ACCESS.2020.2971354.

[51] T. Edgar, D. Manz, Research methods for cyber security, Syngress, 2017.

[52] A. Subasi, Practical Machine Learning for Data Analysis Using Python, Academic Press, 2020.

[53] V. N. Gudivada, M. T. Irfan, E. Fathi, and D. L. Rao, "Cognitive Analytics: Going Beyond Big Data Analytics and Machine Learning," *Handbook of Statistics*, vol. 35, no. 5, pp. 169–205, 2016, doi: 10.1016/bs.host.2016.07.010.

[54] H. Belyadi,A. Haghighat, "Supervised learning," *Machine Learning Guide for Oil and Gas Using Python*, pp. 169-295, 2021, https://doi.org/10.1016/B978-0-12-821929-4.00004-4.

[55] N. K. Chauhan and K. Singh, "A review on conventional machine learning vs deep learning," *2018 International Conference on Computing, Power and Communication*

Technologies, GUCON 2018, no. September 2018, pp. 347–352, 2019, doi: 10.1109/GUCON.2018.8675097.

[56] J. Díaz-Ramírez, "Machine Learning and Deep Learning," *Ingeniare*, vol. 29, no. 2, pp. 182–183, 2021, doi: 10.4067/S0718-33052021000200180.

[57] H. Liang, X. Sun, Y. Sun, and Y. Gao, "Text feature extraction based on deep learning: a review," *Eurasip Journal on Wireless Communications and Networking*, vol. 2017, no. 1, pp. 1–12, 2017, doi: 10.1186/s13638-017-0993-1.

[58] A. K. Sharma, S. Chaurasia, and D. K. Srivastava, "Sentimental Short Sentences Classification by Using CNN Deep Learning Model with Fine Tuned Word2Vec," *Procedia Computer Science*, vol. 167, no. 2019, pp. 1139–1147, 2020, doi: 10.1016/j.procs.2020.03.416.

[59] I. H. Sarker, "Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions," *SN Computer Science*, vol. 2, no. 6, pp. 1–20, 2021, doi: 10.1007/s42979-021-00815-1.

[60] C. Yao et al., "A Convolutional Neural Network Model for Online Medical Guidance," IEEE Access, vol. 4, no. January 2019, pp. 4094–4103, 2016, doi: 10.1109/ACCESS.2016.2594839.

[61] R. K. Mishra, G. Y. S. Reddy, and H. Pathak, "The Understanding of Deep Learning: A Comprehensive Review," *Mathematical Problems in Engineering*, vol. 2021, 2021, doi: 10.1155/2021/5548884.

[62] H. Ahmad, M. Z. Asghar, F. M. Alotaibi, and I. A. Hameed, "Applying Deep Learning Technique for Depression Classification in Social Media Text," Journal of Medical Imaging and Health Informatics, vol. 10, no. 10, pp. 2446–2451, 2020, doi: 10.1166/jmihi.2020.3169.

[63] B. Jang, M. Kim, G. Harerimana, S. U. Kang, and J. W. Kim, "Bi-LSTM model to increase accuracy in text classification: Combining word2vec CNN and attention mechanism," Applied Sciences (Switzerland), vol. 10, no. 17, 2020, doi: 10.3390/app10175841.

[64] G. Chowdhury, "Natural language processing ," *The annual review of Information science and technology*, vol. 37, pp. 51–89, 2003.

[65] A. Tabassum and R. R. Patil, "A Survey on Text Pre-Processing & Feature Extraction Techniques in Natural Language Processing," *International Research Journal of Engineering and Technology*, no. June, pp. 4864–4867, 2020, [Online]. Available: www.irjet.net.

[66] S. Doshi, "*Twitter Sentiment Analysis using fastText,*", Towards Data Science https://towardsdatascience.com/twitter-sentiment-analysis-using-fasttext-9ccd04465597 , (10-Feb-2022) .

[67] H. A. Alatabi and A. R. Abbas, "Sentiment analysis in social media using machine learning techniques," *Iraqi Journal of Science*, vol. 61, no. 1, pp. 193–201, 2020, doi: 10.24996/ijs.2020.61.1.22.

[68] E. J. L. S.Vidhya , D.Asir Antony Gnana Singh, "Feature Extraction for Document Classification," *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 04, no. 06, pp. 50–56, 2016, [Online]. Available: www.ijirset.com

[69] R. Ahuja, A. Chug, S. Kohli, S. Gupta, and P. Ahuja, "The impact of features extraction on the sentiment analysis," *Procedia Computer Science*, vol. 152, pp. 341–348, 2019, doi: 10.1016/j.procs.2019.05.008.

[70] H.Park, K.Kim, "45" *Journal of The Korea Society of Computer and Information*, vol. **25**, pp. 181-188, 2020, doi: https://doi.org/10.9708/jksci.2020.25.08.181.

[71] L. Ma and Y. Zhang, "Using Word2Vec to process big text data," *Proceedings - 2015 IEEE International Conference on Big Data, IEEE Big Data 2015*, no. October 2015, pp. 2895–2897, 2015, doi: 10.1109/BigData.2015.7364114.

[72] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, pp. 1–12, 2013.

[73] Hugging Face, "*Word2Vec,*", hugging Face, https://huggingface.co/fse/word2vec-google-news-300, (21-Nov-2022).

[74] Z. Karimi, "*Confusion Matrix,*", Research gate https://www.researchgate.net/publication/355096788_Confusion_Matrix , (10-Dec-2022) .

[75] Ting, K.M., "Confusion Matrix," in *Title: Sammut*, C., Webb, G.I. (eds) Encyclopedia of Machine Learning. USA: Springer, Boston, MA, 2011, pp. 209, https://doi.org/10.1007/978-0-387-30164-8_157.

[76] D.Yoo,2022, "Depressive tweets propessed", Kaggle, [Online]. Available: https://www.kaggle.com/code/dongphilyoo/detect-early-depression-through-tweets/data?select=depressive_tweets_processed.csv .

[77] S.Sahoo, 2020," DepressionTweets" , Kaggle, [Online]. Available: https://www.kaggle.com/datasets/samrats/depressiontweets?select=test2Data.csv.

[78] M. Al-Mosaiwi and T. Johnstone, "In an Absolute State: Elevated Use of Absolutist Words Is a Marker Specific to Anxiety, Depression, and Suicidal Ideation," Clinical Psychological Science, Vol. 6, No. 4, pp. 529–542, 2018, doi: 10.1177/2167702617747074.

[79] N. Z. Zulkarnain, H. Basiron, and N. Abdullah, "Writing style and word usage in detecting depression in social media: A review," Journal of Theoretical and Applied Information Technology, Vol. 98, No. 1, pp. 124–135, 2020.

[80] F. Boumahdi, A. Madani, and I. Cheurfa, "Identifying Depression in Tweets Using CNN-deep and BILSTM with Attention Model," vol. 12, no. 2, pp. 47–61, doi: 10.6025/ijwa/2020/12/2/47-61.

[81] C. Adamuthe, "Improved Text Classification using Long Short-Term Memory and Word Embedding Technique," International Journal of Hybrid Information Technology, vol. 13, no. 1, pp. 19–32, 2020, doi: 10.21742/ijhit.2020.13.1.03.

Appendix A
The Published Paper

Bulletin of Electrical Engineering and Informatics
Vol. 99, No. 1, Month 2099, pp. 1~1x
ISSN: 2302-9285, DOI: 10.11591/eei.v99i1.paperID

Predicating Depression in Twitter Using Hybrid Deep and Machine Learning Techniques

Rula Kamil[1], Ayad R.Abbas[2]
[1]Department of Software, College of Information Technology, University of Babylon, Iraq
[2] Department of Computer Science, University of Technology, Iraq

Article Info

Article history:

Received month dd, yyyy
Revised month dd, yyyy
Accepted month dd, yyyy

Keywords:

Depression
Twitter
Hybrid model
Bi-LSTM
XGBOOST

ABSTRACT

Nowadays, depression is a common mental illness. Failure to recognize depression early or guarantee that a depressed individual receives prompt counseling can lead to serious issues. Social media allow us to monitor people's thoughts, daily activities, and emotions, including persons with mental illnesses. This study suggested novel hybrid models that combine one of the deep learning techniques with one of the machine learning approaches. This paper used a dataset from the Kaggle website to predict depression. Two deep learning techniques were chosen to conduct the experiments: bidirectional long short-term memory (Bi-LSTM), and convolutional neural network (CNN). Three machine learning techniques were also selected, which are support vector machine (SVM), light gradient boosting machine (LGBM), and extreme gradient boosting (XGBOOST). Deep learning methods were applied to extract important features from input data and training, and then machine learning was utilized to predict the class. The performance of the hybrid models was compared against that of five single models. The results showed that Bi-LSTM-XGBOOST is better than single models and achieve the highest performance, with 94% for all evaluation metrics. The proposed model can improve the performance of machine learning techniques and increase the detection rate of depression.

Corresponding Author:

Rula Kamil
College of Information Technology, Department of Software
University of Babylon
Iraq
Email: rola.kamil@uobabylon.edu.iq

1. INTRODUCTION

Nowadays, depression is a widespread illness. There are many people worldwide who suffer from depression. Around 264 million people of all ages suffer from depression, as indicated by the World Health Organization (WHO) [1]. Depression is one of the most common causes of suicide, with over 800,000 suicide deaths occurring every year, moreover, it is the second leading cause of death among individuals in the 15- to 29-year-old range [2].

Depression is a popular mental disorder [2]. Depressive disorders come in a variety of types. For each type, there are a set of symptoms. Major depressive disease (MDD) is a popular type of depressive disorder. With this type, people cannot sleep, eat, or work. During at least two weeks, each day, the patient must have at least five of the symptoms. The symptoms include a sad mood for most of the day, a lack of interest in all activities, losing or gaining weight, a lack of energy, body agitation or retardation, a feeling of remorse or worthlessness, an inability to sleep or sleep for longer periods, and thoughts of death and suicide [3]. There are many methods used in medical fields to diagnose depression, such as surveys and interviews, but they are

Predicting Depression on Twitter Using Machine Learning Techniques

Rula Kamil [1]*, **Ayad R.Abbas** [2]
[1] *Department of Software, College of Information Technology, University of Babylon, Babylon, Iraq*
[2] *Department of Computer Science, University of Technology, Baghdad, Iraq*

Abstract

Depression has grown to be a common mental disorder worldwide, and a suicidal attempt are a common complication among depressed people. Despite the broad health care programs and treatment availability, the detection rate remains rather low, and most depressed people refuse to look for treatment. Social media has become an essential part of people's lives where they share daily activities, thoughts, and feelings with others. In addition, nowadays, machine learning techniques and deep learning are widely used in sentiment analysis. Therefore, it is easy to utilize them for the early detection of depression. This paper will focus on three main objectives. The first is developing a classifier model for the early detection of depression among Twitter users. The second is studying the impact of some pre-processing steps on the performance of classifiers that were ignored in previous studies. The third is comparing the performance of different machine learning techniques and pre-processing. Support Vector Machine (SVM), Logistic Regression (LR), and Light GBM (LGBM) as three machine learning approaches are applied. The results indicate that the best accuracy obtained is 93% with the application of stemming, replacing slang words and emoji, and not removing some stop words with SVM technique.

Keywords: Depression, Machine learning, Social media, Twitter, SVM.

1.Introduction

Nowadays, depression has become one of the most common mental illnesses worldwide, as there are 300 million people are affected by it. During a depressive episode, the patient loses interest or pleasure in daily activity and suffers from tiredness, lack of sleep, eating disorders, hopelessness, and even suicidal ideation [1]. Approximately eight million people die every year due to mental disorders [2], which is the same number of patients who die every year due to cancers [3]. Despite health care programs and treatment protocols availability, however, 50% of potential cases of depression go undetected in high-income countries, with eventually higher percentages in middle and low-income countries [4]. The psychiatrist can assess the mental health condition of people through their language, because language reveals what people think and feel. Therefore, users of social media who suffer from depression can be detected by assessing what they write, especially the negative emotional words [5]. 'Depression language' may have a strong effect on its writers, suicidal poets like Sylvia Plath and Kurt Cobain used suicidal words associated with self in their poetries, and then they eventually killed themselves [6]. Depression can be diagnosed by face-to-face medical interview, but 70% of the patients would not consult doctors [7]. It could also be diagnosed via a 'Patient Health Questionnaire', but the patient may not answer correctly [8,9]. The third way could be by asking friends and relatives of the patient, however this remains and inaccurate method [10]. Billions of people around the world are users of

الخلاصة

في الوقت الحاضر، يعد الاكتئاب مرضًا عقليًا شائعًا. يمكن أن يؤدي الفشل في التعرف على الاكتئاب مبكرًا أو ضمان تلقي الفرد المكتئب للاستشارة الفورية إلى مشاكل خطيرة. تسمح لنا وسائل التواصل الاجتماعي بمراقبة أفكار الناس وأنشطتهم اليومية وعواطفهم ، بما في ذلك الأشخاص المصابون بأمراض عقلية. في الوقت الحاضر ، تُستخدم تقنيات التعلم الآلي والتعلم العميق على نطاق واسع في تحليل المشاعر. لذلك ، من السهل استخدامها للكشف المبكر عن الاكتئاب.

ستركز هذه الأطروحة على ثلاثة أهداف رئيسية. الأول هو تطوير نموذج مصنف للكشف المبكر عن الاكتئاب بين مستخدمي تويتر. والثاني هو دراسة تأثير بعض خطوات ما قبل المعالجة على أداء المصنفات التي تم تجاهلها في الدراسات السابقة. والثالث هو مقارنة أداء تقنيات التعلم الآلي المختلفة وإيجاد أفضل التقنيات.

لتحقيق هذه الأهداف، تم تنفيذ ثلاثة أنواع من تقنيات التعلم الآلي في هذه الأطروحة. يتضمن النوع الأول خمسة تعلم آلي تقليدية

1) support vector machine (SVM)
2) light gradient boosting machine (LGBM)
3) extreme gradient boosting (XGBOOST)
4) logistic regression (LR)
5) decision tree (DT)

النوع الثاني يتضمن اثنين من تقنيات التعلم العميق :

1. bidirectional long short-term memory (Bi-LSTM).
2. convolutional neural network (CNN).

النوع الثالث هو نموذج هجين يجمع بين إحدى تقنيات الآلة التقليدية وأحد تقنيات التعلم العميق.
تم استخدام مجموعتين من البيانات من موقعKaggle للتنبؤ بالاكتئاب.

أظهرت التجارب على مجموعة البيانات الأولى أن Bi-LSTM-XGBOOST :أفضل من النماذج الفردية ويحقق أعلى أداء ، بنسبة 94٪ لجميع مقاييس التقييم. يمكن للنموذج المقترح تحسين أداء تقنيات التعلم الآلي وزيادة معدل اكتشاف الاكتئاب. ثانيًا ، يمكن أن يؤدي تطبيق Stemming والاستبدال

(الكلمات العامية والرموز التعبيرية) وعدم إزالة بعض الكلمات المتوقفة إلى تحسين دقة تقنيات التعلم الآلي التقليدية.

بينما توضح مجموعة البيانات الثانية أن LGBM تفوق على التقنيات الأخرى بنسبة 99٪ لجميع مقاييس التقييم.

كبر حجم مجموعة البيانات الثانية كان له دورا مهما في تفوق الحصول على دقة اعلى من مجموعة البيانات الاولى.

جمهورية العراق
وزارة التعليم العالي والبحث العلمي
جـامعـــة بـابـــل
كلية تكنولوجيا المعلومات
قسم البرمجيات

التنبؤ بالاكتئاب لدى مستخدمي وسائل التواصل الاجتماعي باستخدام تقنيات تعلم الالة

رسالة مقدمة إلى
مجلس كلية تكنولوجيا المعلومات - جامعة بابل وهي جزء من متطلبات
نيل درجة الماجستير في تكنولوجيا المعلومات / البرمجيات

من قبل

رولا كامل حسن عباس

بإشـــراف

ا . د . اياد روضان عباس

1444 هـ

2023 م