

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B. Tech. Final Year. We owe special debt of gratitude to Professor Dr. Sanjiv Sharma, Department of Computer Science & Engineering, KIET, Ghaziabad, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavours have seen light of the day.

We also take the opportunity to acknowledge the contribution of Dr. Vineet Sharma, Head of the Department of Computer Science & Engineering, KIET, Ghaziabad, for his full support and assistance during the development of the project. We also do not like to miss the opportunity to acknowledge the contribution of all the faculty members of the department for their kind assistance and cooperation during the development of our project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members, especially Professor Ankur Bhardwaj, of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Date:

Signature:

Name: Apurva Goel

Roll No.: 1900290100036

Signature:

Name: Mirza Tabish Hasan

Roll No.: 2000290109011

Signature:

Name: Nazakat Ali Sofi

Roll No.: 1900290100089

ABSTRACT

Social media has appeared as one of the platforms to uplift user's opinions. Twitter sentiments analysis is the method of computation of user opinions. Twitter is one of the social media that is gaining popularity. Twitter offers organizations a fast and effective way to analyse customers' perspectives toward the critical to success in the marketplace. Developing a program for sentiment analysis is an approach to be used to computationally measure customers' perceptions. There are three types of sentiments: Positive, Neutral and Negative. Analysing the user sentiments was the biggest problem in the early days but now it can be solved with the help of Machine Learning Algorithms. Twitter is an online micro-blogging and social-networking platform which allows users to write short status updates and through the Tweets are helpful in extracting the Sentimental values from the user. It is focused on the person's tweets and the hash tags for understanding the situations in each aspect of the criteria. This can help us to develop business strategies and understand customer's feeling towards product or brand or we can know the sentiments behind the opinion on any topic on which the group of people tweet. In this project, the tweets collected from several events, analysed them using nine Machine Learning algorithms like SVM with different kernels (i.e., sigmoid, polynomial, radial bias function and linear), Decision Tree, Random Forest, KNN, GBM and Logistic Regression and compared the results.

Keywords: Machine Learning, Microblogging, Twitter, NLP, Classification algorithm, Python

TABLE OF CONTENTS	Page No.
DECLARATION.....	ii
CERTIFICATE.....	iii
ACKNOWLEDGEMENTS.....	iv
ABSTRACT.....	v
LIST OF FIGURES.....	ix
LIST OF TABLES.....	xi
LIST OF ABBREVIATIONS.....	xii
CHAPTER 1 (Introduction)	1
1.1. Introduction.....	1
1.2. Project Description.....	2
1.3. Report Structure.....	2
1.4. Summary.....	4
CHAPTER 2 (Literature Review)	5
2.1. Introduction.....	5
2.2. Machine learning.....	5
2.2.1. Supervised learning	6
2.2.2. Unsupervised learning	9
2.2.3. Semi-supervised learning	11
2.2.4. Reinforcement learning	11

2.3. Logistic Regression.....	13
2.4. Bayesian Belief Network.....	14
2.5. Decision Tree.....	15
2.6. Support Vector Machine.....	17
2.6.1. Types of SVM.....	19
2.6.2. Kernel Function.....	19
2.6.3. Major Kernel Functions.....	19
2.7. KNN Algorithm.....	21
2.7.1. Need of KNN.....	22
2.7.2. Working of KNN.....	23
2.7.3. Selection of K in the KNN Algorithm.....	25
2.8. Random Forest Algorithm.....	26
2.8.1. Working.....	27
2.9. Gradient Boosting Machine.....	27
2.9.1. Working.....	28
2.9.2. Loss Function.....	29
2.9.3. Weak Learner.....	29
2.9.4. Additive Model.....	31
 CHAPTER 3 (PROPOSED METHODOLOGY)	32
3.1. Data Gathering	32
3.2. Dataset Without Removing Punctuation	33

3.3. Natural Language Processing (NLP)	33
3.4. Pre Processing	34
3.5. Dataset After Cleaning	34
3.6. Model Training	35
CHAPTER 4 (RESULTS AND DISCUSSION)	37
CHAPTER 5 (CONCLUSIONS AND FUTURE SCOPE)	40
5.1. Conclusion.....	40
5.2. Future Scope.....	40
REFERENCES.....	42

LIST OF FIGURES

Figure No.	Description	Page No.
1.1	Report Summary	3
2.1	Types of Learning	6
2.2	Supervised Learning	8
2.3	Unsupervised Learning	10
2.4	Logistic Function	13
2.5	Decision Tree	16
2.6	SVM	18
2.7	Working of SVM	19
2.8	Sigmoid Kernel Graph	20
2.9	Polynomial Kernel Graph	21
2.10	Gaussian Kernel Graph	21
2.11	KNN Classifier	22
2.12	Category Graph	23
2.13	KNN Working	24
2.14	Euclidean Distance	24
2.15	KNN Graph	25
2.16	Random Forest Working	26
3.1	Sample of Data	32
3.2	Graphical Representation of Data	33
3.3	Dataset Before Cleaning	33

3.4	Data After Cleaning	35
4.1	Accuracy Graph	39
4.2	Confusion Matrix	39

LIST OF TABLES

Table. No.	Description	Page No.
2.1	Supervised Learning	7
4.1	Performance Parameter	38
4.2	Accuracy Comparison	38

LIST OF ABBREVIATIONS

1. NLP	Natural Language Processing
2. SVM	Support Vector Machines
3. ML	Machine Learning
4. AI	Artificial Intelligence
5. ANN	Artificial Neural Network
6. RNN	Recurrent Neural Network
7. CNN	Computational Neural Network
8. KNN	K- Nearest Neighbour
9. RBF	Radial Basis Function
10. GBM	Gradient Boosting Machine

CHAPTER 1

Introduction

1.1 Introduction

Written text, spoken language, sign languages, and other forms of communicative media all convey some sort of objective or subjective information. Sentiment is the collective term for the intangible impressions such as attitudes, feelings, views, assessments, and opinion that are connected to any kind of communication. Sentiment analysis, often known as opinion mining, is a branch of study with the goal of categorising the sentiments connected to any spoken or written language. Sentiment analysis is a branch of study that determines if and to what extent the content of spoken or written words is pleasant, unfavourable, or neutral. It does this by using Natural Language Processing (NLP), machine learning, statistics, linguistic aspects, etc. There are several uses for sentiment analysis. An important tool for information extraction is sentiment analysis.

Websites for microblogging are among the most significant sources of information. This is because everyone expresses their thoughts on a wide range of topics, talks about current events, moans, and gives good feedback on products they use on a regular basis. Sentimental analysis is the method used to extract reliable information from texts. It is the process of creating structured data from unstructured data, to put it another way. This is used to gauge consumer thoughts, feedback, and product reviews. Unstructured data includes information from the internet, such as chats, emails, pdfs, word documents, e-commerce websites, and social networking sites. Sentiment analysis is contextual mining of words which shows the social sentiment of the data.

The number of people expressing their views and opinions online is growing along with the rapid advancement of web technologies. Everyone can benefit from this information, including individuals, corporations, and governments. Twitter is evolving into a significant information source with its 500+ million tweets every day. Twitter is a microblogging platform most known for its 140-character messages, or tweets. There is a 140-character restriction. Twitter is a good resource for information because it has 240+ million active users. Users frequently talk about their personal opinions on many topics and current events in tweets. We

chose Twitter out of all the main social media platforms, including Facebook, Twitter, Google+, and Myspace.

1.2 PROJECT DESCRIPTION

In this project we have used different machine learning models to analyse the sentiments of twitter. To evaluate the better result there are two main factors such as quality of data set and quantity of dataset. The data set is taken from the Kaggle. It consist of 1.6 Million tweets. This data set have six columns (i.e., Target, Ids, Date, Flag, User, Tweet). 80% of the data set is taken as training data set and rest 20% is testing data set. In the project the accuracy of prediction is calculated. It also provides the f1-score. There are three types of sentiments: Positive, Neutral and Negative. These are the following machine learning algorithms which are used during this project:

1. Naïve Bayes
2. Support Vector Machine with different kernels (i.e., RBF, Sigmoid, Polynomial, Linear)
3. Decision Tree
4. Random Forest Algorithm
5. KNN
6. Gradient Boosting Machine

These are the steps used in Machine Learning for analysing the sentiment:

- Stemming
- Tokenization
- Part of speech tagging
- Parsing
- Lexicon analysis (depending on the relevant context)

1.3 Report Structure

This report paper has five chapters that include a preface, followed by a literature review,

Exploration methodology, data findings, and analysis and the last bone is a conclusion with suitable recommendations as shown in figure 1.1.

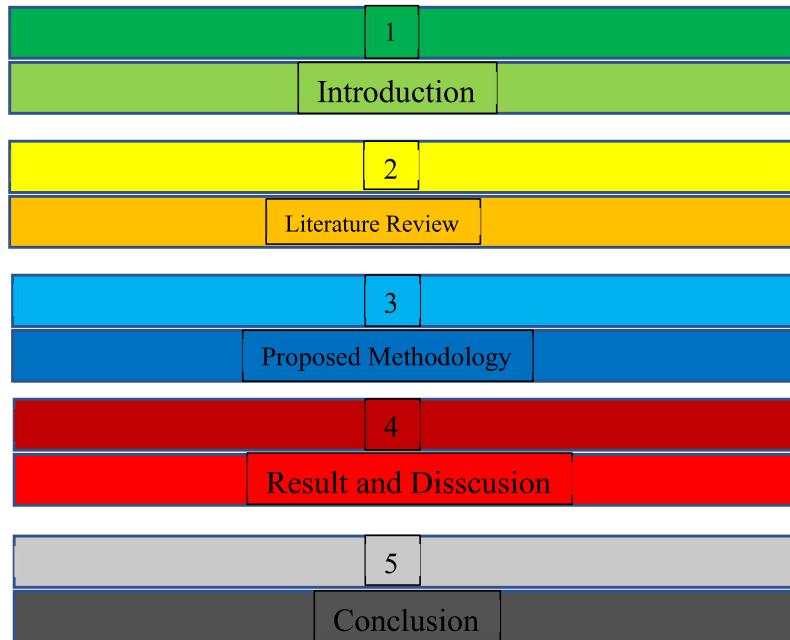


Figure 1.1: Report Summary

The first chapter provides a transparent summary of this project called A Machine Learning Approach of Sentiment Analysis on Twitter. The explicit chapter, therefore, works to highlight that how different machine learning approaches works. In addition to it, a summary of how the prediction of sentiments will happen. It provides us a complete overview of tweets and sentiment analysis. This tells that data set should have good quality and quantity. A good result requires good data set for training.

The second chapter is the literature review that does work on finding the research variables with the use of definitions, an overview, and the past content. The section works on the various sentiment analysis prediction methodologies and allows a clear understanding of many algorithms used for it. Additionally, the second chapter aims to critically appraise the impact of analysis variables on each other. This chapter also talks about the step-by-step process of many machine learning algorithms used for sentiment analysis.

The third chapter is the methodology; during this chapter, the algorithms are implemented for the sentiment analysis. At this step all the algorithms work on the data set and provide their accuracy and f1-score. In this chapter all the nine ways are implemented. The analysis makes use of surveys to gather information and analyse them using Machine Learning.

The fourth chapter is s and discussion, the collected data from primary sources area units analysed through the comparative analysis of algorithms. The illustrate that which algorithm is better for the given data set and what is their accuracy and f1-score. The discussion of result and a comparative analysis of different methodologies is done.

The fifth chapter conclude that the outcome of the analysis paper is terminated, and suggestion will be enforced to enhance this analysis shortly.

1.4 Summary

This chapter introduces the topic of this report and has mentioned the reasons why we research this topic in detail. Background problems related to this research topic have been discussed in this chapter of the dissertation. On the other hand, the aim, objectives, and questions of this research are postulating the purpose of this research. This work is critical to review the problems associated with heart condition prediction and to seek out a far better solution by implementing an appropriate technique into the prediction model.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

Machine Learning Classification and Bayesian networks are some of the core concepts and terminologies covered in this chapter. Related and existing works on driver drowsiness detection using various machine learning techniques such as Naive Bayes. Logistic Regression. KNN. Decision Tree. SVM, and others are examined, with an emphasis on what was done, how it was done the classification technique used, the data set used for implementation, the tools used, and the system's result and accuracy.

2.2 Machine Learning

Since 1959, the field of Machine Learning has existed. While working for IBM. Arthur Samuel described Machine Learning as a branch of research that allows computers to learn without having to be explicitly taught. Various strategies and procedures are used in all sectors to obtain relevant knowledge from unprocessed real-world data.

We create a hybrid strategy in this thesis that combines classic statistical methods with model-based machine learning techniques to efficiently identify driver drowsiness detection systems with smaller feature sets and higher accuracy. In this thesis, we propose a hybrid strategy that combines classic statistical methods with model-based machine learning techniques to identify driver drowsiness detection with smaller feature sets and higher accuracy.

Machine Learning has been a popular field of Artificial Intelligence (AI) study and application for decades. Machine Learning is being applied in self-driving cars, speech recognition, effective web search, and facial detection, to name a few applications. Machine learning algorithms are divided into four categories as shown in figure 2.1:

- Supervised Learning Algorithm
- Unsupervised Learning Algorithm

- Semi-Supervised Learning
- Reinforcement Learning

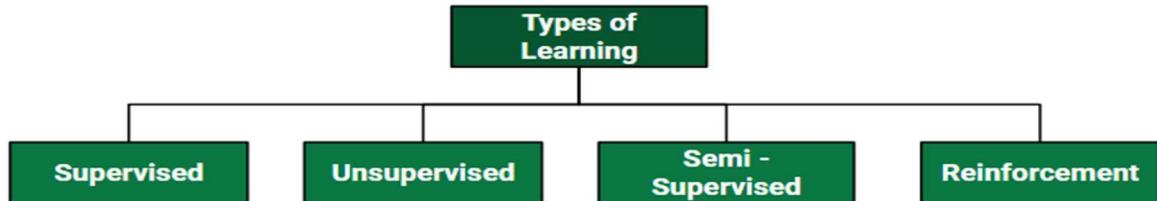


Figure 2.1: Types of Learning

2.2.1 Supervised Learning

Supervised learning is when the model is getting trained on a labelled dataset. A labelled dataset is one that has both input and output parameters. In this type of learning both training and validation, datasets are labelled as shown in the figures below. Supervised learning is the types of machine learning in which machines are trained using well "labelled" training data, and on basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output. In supervised learning, the training data provided to the machines work as the supervisor that teaches the machines to predict the output correctly. It applies the same concept as a student learns in the supervision of the teacher. Supervised learning is a process of providing input data as well as correct output data to the machine learning model. The aim of a supervised learning algorithm is to find a mapping function to map the input variable(x) with the output variable(y).

Supervised learning is the types of machine learning in which machines are trained using well "labelled" training data, and on basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output. In supervised learning, the training data provided to the machines work as the supervisor that teaches the machines to predict the output correctly.

Table 2.1: Supervised Learning

User ID	Gender	Age	Salary	Purchased	Temperature	Pressure	Relative Humidity	Wind Direction	Wind Speed
15624510	Male	19	19000	0	10.69261758	986.882019	54.19337313	195.7150879	3.278597116
15810944	Male	35	20000	1	13.59184184	987.8729248	48.0648859	189.2951202	2.909167767
15668575	Female	26	43000	0	17.70494885	988.1119385	39.11965597	192.9273834	2.973036289
15603246	Female	27	57000	0	20.95430404	987.8500366	30.66273218	202.0752869	2.965289593
15804002	Male	19	76000	1	22.9272874	987.2833862	26.06723423	210.6589203	2.798230886
15728773	Male	27	58000	1	24.04233986	986.2907104	23.46918024	221.1188507	2.627005816
15598044	Female	27	84000	0	24.41475295	985.2338867	22.25082295	233.7911987	2.448749781
15694829	Female	32	150000	1	23.93361956	984.8914795	22.35178837	244.3504333	2.454271793
15600575	Male	25	33000	1	22.68800023	984.8461304	23.7538641	253.0864716	2.418341875
15727311	Female	35	65000	0	20.56425726	984.8380737	27.07867944	264.5071106	2.318677425
15570769	Female	26	80000	1	17.76400389	985.4262085	33.54900114	280.7827454	2.343950987
15606274	Female	26	52000	0	11.25680746	988.9386597	53.74139903	68.15406036	1.650191426
15746139	Male	20	86000	1	14.37810685	989.6819458	40.70884681	72.62069702	1.553469896
15704987	Male	32	18000	0	18.45114201	990.2960205	30.85038484	71.70604706	1.005017161
15628972	Male	18	82000	0	22.54895853	989.9562988	22.81738811	44.66042709	0.264133632
15697686	Male	29	80000	0	24.23155922	988.796875	19.74790765	318.3214111	0.329656571
15733883	Male	47	25000	1					

Figure A: CLASSIFICATION

Figure B: REGRESSION

Both the above figures have labelled data set as follows:

Figure A: It is a dataset of a shopping store that is useful in predicting whether a customer will purchase a particular product under consideration or not based on his/ her gender, age, and salary.

Input: Gender, Age, Salary

Output: Purchased i.e. 0 or 1; 1 means yes the customer will purchase and 0 means that the customer won't purchase it.

Figure B: It is a Meteorological dataset that serves the purpose of predicting wind speed based on different parameters.

Input: Dew Point, Temperature, Pressure, Relative Humidity, Wind Direction

Output: Wind Speed

Training the system:

While training the model, data is usually split in the ratio of 80:20 i.e., 80% as training data and the rest as testing data. In training data, we feed input as well as output for 80% of data. The model learns from training data only. We use different machine learning algorithms (which we will discuss in detail in the next articles) to build our model. Learning means that the model will build some logic of its own. Once the model is ready then it is good to be tested. At the time of testing, the input is fed from the remaining 20% of data that the model has never seen before, the model will predict some value and we will compare it with the actual output and calculate the accuracy.

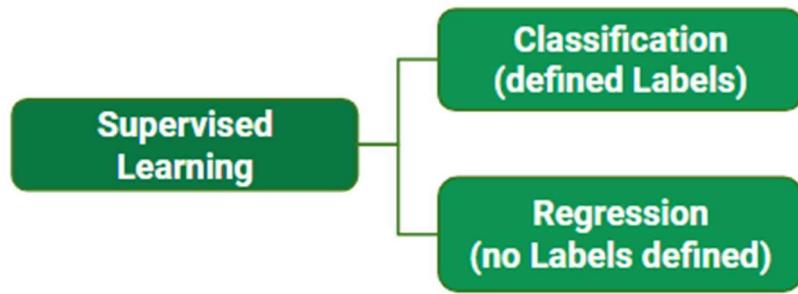


Figure 2.2: Supervised Learning

Types of Supervised Learning:

A. Classification: It is a Supervised Learning task where output is having defined labels (discrete value). For example, in above Figure A, Output – Purchased has defined labels i.e., 0 or 1; 1 means the customer will purchase, and 0 means that the customer won't purchase. The goal here is to predict discrete values belonging to a particular class and evaluate them based on accuracy.

It can be either binary or multi-class classification. In binary classification, the model predicts either 0 or 1; yes or no but in the case of multi-class classification, the model predicts more than one class. Example: Gmail classifies mails in more than one class like social, promotions, updates, and forums.

B. Regression: It is a Supervised Learning task where output is having continuous value. For example, in above Figure B, Output – Wind Speed is not having any discrete value but is continuous in a particular range. The goal here is to predict a value as much closer to the actual output value as our model can and then evaluation is done by calculating the error value. The smaller the error the greater the accuracy of our regression model.

The classification based on supervised learning approaches is shown in Figure 2.2

Example of Supervised Learning Algorithms:

- Linear Regression
- Logistic Regression

- Nearest Neighbour
- Gaussian Naive Bayes
- Decision Trees
- Support Vector Machine (SVM)
- Random Forest

2.2.2 Unsupervised Learning

To anticipate the outcome from unlabelled datasets, unsupervised learning is utilized. The most frequent unsupervised learning technique is clustering. Consider a scenario; the unsupervised system is given an input dataset that contains the photographs of various cats and dogs.

Because the algorithm is never trained on the given dataset, it has no idea what its characteristics are. The purpose of the unsupervised learning algorithm is to recognize visual elements independently. This task will be done by dividing the image collection into groups based on image similarity using an unsupervised learning method. The following are some of the most important arguments for the relevance of unsupervised learning:

- Unsupervised learning is beneficial for extracting relevant information from data.
- Unsupervised learning is analogous to how a human learns to think via their own experiences, bringing it closer to true AI.
- Because unsupervised learning works with unlabelled and uncategorized data, it is more important.
- In the real world, we don't always have input data that corresponds to output, hence we require unsupervised learning to handle these problems. Some unsupervised learning algorithms are listed below:
 - K-means clustering KNN (k-nearest neighbours)
 - Hierarchical clustering
 - Anomaly detection Neural Networks
 - Principal Component Analysis

The classification based on unsupervised learning approaches is shown in Figure 2.3

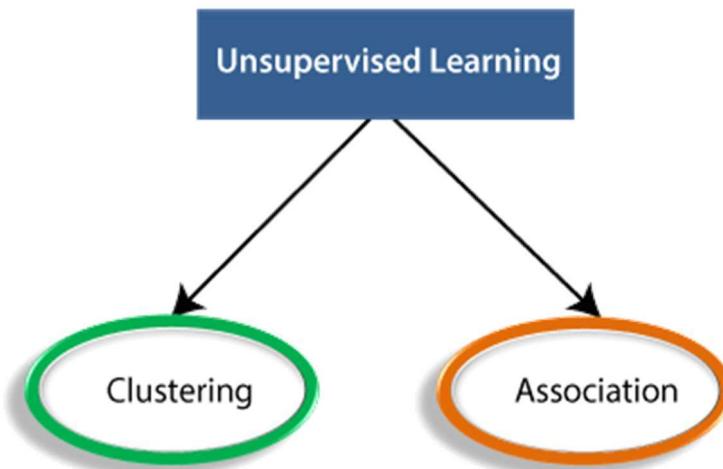


Figure 2.3: Unsupervised Learning

Clustering

It's essentially an unsupervised learning technique. Unsupervised learning is a technique for extracting references from datasets that contain input data but no labelled answers. It's a method for identifying significant structures, explaining underlying processes, and generating traits, and groups in a set of samples. Clustering is the practice of dividing a population or set of data points into various groups so that data points in the same group are more similar and data points in other groups are more dissimilar... It is essentially a collection of objects based on their similarity and dissimilarity.

Association

Association rule learning is an unsupervised learning technique that examines the dependency of one data item on another and maps accordingly to make it more profitable. It looks for intriguing relationships or associations between the dataset's variables. It finds interesting links between variables in a database using a set of criteria. Association rule learning, which is utilized in Market Basket analysis, Web usage mining, continuous manufacturing, and other applications, is one of the most important issues in machine learning. Market basket

analysis is a technique used by many large retailers to uncover product relationships. We may comprehend it by using the example of a supermarket, where all things purchased together are grouped.

2.2.3 Semi-Supervised Learning

Semi-supervised learning is similar to supervised learning in that it uses both labelled and unlabelled data as seen in figure 2.4. Because unlabelled data is less expensive and easier to obtain, it uses a little amount of labelled data with a large volume of unlabelled data. Classification, regression, and prediction can all be employed with this form of learning. When the expense of labelling is too high to allow for a fully labelled training procedure, semi-supervised learning comes in handy. Identifying a person's face on a webcam is an curly example of this. It would be nearly impossible to find a big number of tagged text documents, in this case, therefore semi-supervised learning is suitable. This is because having someone read through full-text documents merely to assign a simple classification is inefficient. As a result, semi-supervised learning enables the algorithm to learn from a small number of labelled text documents while classifying many unlabelled text documents in the training set.

By employing pseudo labelling, semi-supervised learning can train the model with less labelled training data than supervised learning. Many neural network models and training methods can be combined in this way. The following is how it works: Just like in supervised learning, train the model with a limited amount of labelled training data until it produces good results. Then use it to forecast the outputs using the unlabelled training dataset, which are faux labels because they may not be completely accurate. Connect the labels from the labelled training data to the pseudo labels you made before. Connect the data inputs in the labelled training data to the data inputs. Then, to reduce error and enhance model accuracy, train the model in the same way you did with the labelled set at the start.

2.2.4 Reinforcement Learning

When an agent learns through trial-and-error interactions with a dynamic environment, this is known as reinforcement learning. It is strongly related to statistics' decision theory, game

theory, and engineering's control theory. The purpose of the machine is to learn to act in such a way that it reduces punishments while maximizing future rewards over its lifetime.

Machine learning includes reinforcement learning. It's all about taking the right steps to maximize your reward in each situation. It is used by a variety of software and computers to determine the best feasible action or path in each situation. Reinforcement learning differs from supervised learning in that supervised learning includes the solution key, allowing the model to be trained with the right answer, whereas reinforcement learning does not. Instead, the reinforcement agent selects what to do to complete the job. It is obligated to learn from its experience in the absence of a training dataset.

2.2.4.1 Main Points in Reinforcement Learning

- Input: The input should be a starting state for the model to work from.
- Output: There are numerous possible outputs, just as there are numerous solutions to a given problem.
- Training: The model will return to a state after training, and the user will decide whether to reward or punish the model depending on its output.
- The model is always evolving.
- The optimal solution is determined based on the maximal reward.

Two types of Reinforcement Learning:

1. Positive Reinforcement: Positive reinforcement learning entails doing something to increase the likelihood of the desired behaviour occurring again. It improves the agents' behaviour and increases the strength of the conduct. This form of reinforcement can last a long period, but too much positive reinforcement might result in an overload of states, which can lessen the consequences.

2. Negative Reinforcement: The opposite of positive reinforcement learning, bad reinforcement learning the likelihood of the given behaviour reoccurring by avoiding the unfavourable scenario. Depending on the situation and conduct, it may be more successful than positive reinforcement, although it only offers reinforcement for the bare minimum of activity.

2.3 Logistic Regression

Harrell et al. (2001) proposed a logistic regression model. It's a statistical data analysis method that works with binary dependent variables. Using logistic regression techniques, the logistic model parameter is estimated. A logistic model is a linear combination of independent variables that determines the likelihood of an event. It is not a classification model in general; rather, it models the output probability using the provided inputs. By adjusting the cut-off values, it is frequently used as a classifier. As a result, the variables below the cut-off values belong to one class, while those above the cut-off values belong to another. The two most common types of logistic regression are multinomial and ordinal logistic regression.

Multinomial logistic regression works with absolute values and divides output values into more than two groups. Ordinal logistic regression is the process of sorting the various outputs produced by a multinomial regression model. The techniques of logistic regression. A logistic model is a linear combination of independent variables that calculates the likelihood of an event. It is not a classification model in general; rather, it models the output probability using the provided inputs. By adjusting the cut-off values, it is frequently used as a classifier. It is not a classification model in general; rather, it models the output probability using the provided inputs. By adjusting the cut-off values, it can be utilized as a classifier. As a result, the variables below the cut-off values belong to one class, while those above the cut-off values belong to another.

Figure 2.4 shows the logistic function.

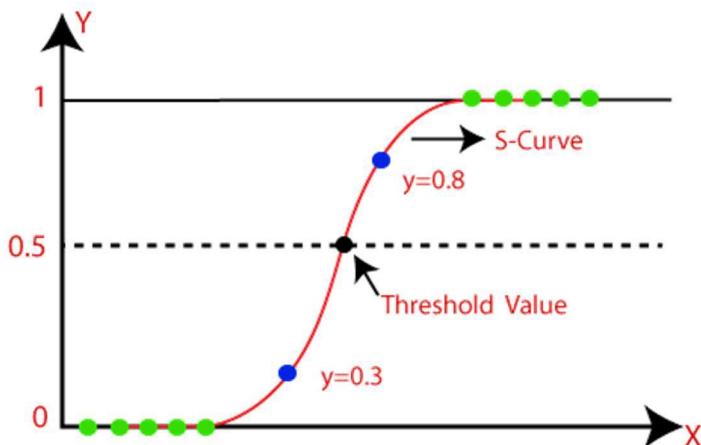


Figure 2.4: Logistic Function

2.3.1 Logistic Regression Equation

The Linear Regression equation can be used to calculate the Logistic Regression equation. Mathematical steps to obtain Logistic Regression equations:

1. Equation for the straight line can be written as:

$$y = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_3 + \dots + b_n x_n \quad (1)$$

2. Because y in Logistic Regression can only be between 0 and 1, divide the previous equation by $(1-y)$:

$$\frac{y}{1-y} : 0 \text{ for } y=0, \text{ and infinity for } y=1 \quad (2)$$

3. However, we require a range of infinity to [infinity, in which case the logarithm of the equation is:

$$\log\left[\frac{y}{1-y}\right] = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_3 + \dots + b_n x_n \quad (3)$$

Above is the final equation of logistic regression.

2.4 Bayesian Belief Network

Bayesian networks are the most well-known classifier for providing a simple and understandable probability distribution (Witten & Frank, 2005). Bayesian network structure is a probabilistic graphical model of some sort. There are nodes and edges in this directed acyclic graph. The edges represent causality, whereas the nodes represent random variables (Spirtes, Glymour, & Scheines, 2001). A conditional probability distribution (CPD) for each node depicts the relationship between the node and its parents.

A direct acyclic graph describes the dependency/independency structure when the joint probability distribution is factorized as the product of numerous conditional distributions (DAG).

$$P(X_1, \dots, X_n) = \prod P(X_i | \text{Pa}(X_i)) \quad (4)$$

X_i 's parent nodes are denoted by $\text{Pa}(X_i)$. The chain rule is another name for this equation. In circumstances where the qualities are interrelated, the conditional independence assumption made by naive Bayes can be stiff. As a result, a more flexible modeling method is required: the Bayesian Belief Network. We specify which attribute pairs are conditionally independent in

BBN. The Bayesian network is a well-established paradigm for managing uncertainty in Artificial Intelligence, and it represents the relationship between nodes using graph theory and probability theory. BN modeling is a type of data mining and machine learning technique that incorporates probabilistic effects generated by large data sets. They are a powerful knowledge representation and an effective tool in uncertain situations.

2.4.1 Some Basic Definitions of Bayesian Belief Network

In a BN, a path between two vertices X and Y is blocked if it travels through a vertex Z and either:

- The connection is either serial ($(X \rightarrow Z \rightarrow Y)$ or $(X \leftarrow Z \leftarrow Y)$) or divergent ($X \leftarrow Z \gg Y$) and Z is conditioned.
- The connection is convergent ($X \rightarrow Z \rightarrow Y$) and neither X nor of its descendants has been influenced by D-separation.

If all pathways from a vertex of X to a vertex of Y are blocked, X and Y are d-separated by 7. If Z divides X and Y in d, then X is independent of Y.

2.5 Decision Tree

These learning methods are commonly utilized in the healthcare industry. Figure 2.5 depicts the decision tree types model. A root node, branches, and leaf nodes are all present in each decision tree. The root node is at the top, while the remaining nodes are called leaf or branch nodes. On one or more properties of the given data, the internal node applies the decision rule or a test. The output is defined by the branch node.

Decision trees are a popular categorization approach since they don't require any prior knowledge of data distribution. It also performs effectively with noisy and ambiguous data. It's been utilized in eHealth applications to categorize patients according to their illnesses. It also uses symptoms to anticipate a patient's sickness. Classification is the process of accurately classifying input data and mapping it to its appropriate classes in predictive analysis. Labelled and unlabelled data are the two most categories of data. There are several predictor attributes and a single target attribute in the labelled data. The class label is represented by each value of the target characteristics. Only the predictor attributes are present in the unlabelled attributes. Now

you might be wondering why we chose the Decision Tree Classifier over other classifiers. We can give two reasons to answer the question. One example of an algorithm is decision trees, which attempt to replicate the way the human brain thinks so that it is relatively straightforward to comprehend the data and reach sound conclusions. Or interpretations. Second, rather than being a black box algorithm like SVM. NN, and others, decision trees allow us to understand the reasoning for the data to interpret. It has the advantage of being straightforward, making it one of the most popular among programmers of this generation.

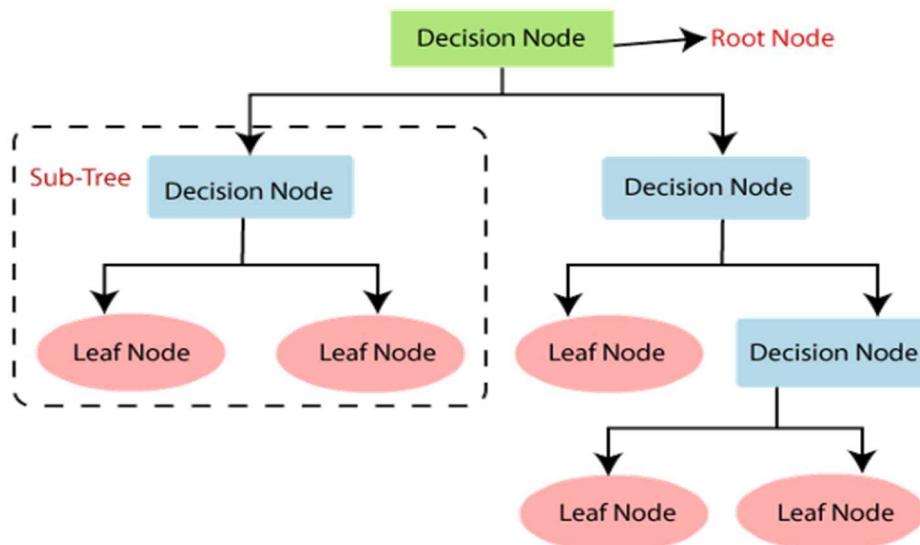


Figure 2.5: Decision Tree

Now that we've established why a Decision Tree is beneficial, let's take a closer look at what a Decision Tree Classifier is. To begin, a decision tree is a tree with several nodes, each of which represents a feature (attribute), each link (branch) represents a decision (rule), and each leaf of the tree represents an outcome (categorical or continuous value). The idea is to make a tree out of all of the data and receive a result at each leaf. We now have a better understanding of what a decision tree is. Let's get started talking about how to make a decision tree classifier. Two alternative algorithms can be used to create a decision tree. CART (Classification and Regression Trees) is one, and ID3 is another (iterative Dichotomiser 3). For ID3, we start with the column's x value and a y value that stays at the column's last position and only has "YES" or "NO" values. The x values for the chart above are (outlook, temp, humidity, and windy) and the value is play,

which only has two options: "YES" or "NO" and is at the bottom of the column. Now we must map the x and y coordinates. Because this is a binary classification problem, we'll use the ID3 technique to construct the tree.

As a general guideline, the root node should be the characteristic that has the most impact on the value y. Then we move on to the next node, which is the most influential feature. We'll employ the concept of entropy, which is a measure of how much uncertainty there is in a data set. For the binary classification task, we must calculate the entropy for all categorical values. To summarize, we must first compute the entropy for the data set. Then we need to choose the highest gain attribute and repeat it till we have the tree we want. That is the ID3 process now. As previously stated, the Decision Tree Classifier is based on a different method known as CART, which stands for classification and regression trees. We utilize the Gini Index as our cost function to evaluate splits in the dataset in our approach. Because our target variable is binary, it can only have two values (yes and no).

Now we must calculate the Gini score, which will help us determine how to divide the data. If the Gini score is 0, we can consider it a perfect separation, while a 50/50 split is the worst-case scenario. The difficulty now is how to determine the Gini index value. The Gini index will remain identical if the target variable is a category variable with numerous levels. The initial step in this procedure is to compute the gain index for the data-set. Then we must calculate the Gini index for all category values, determine the average information entropy for the current attribute, and finally calculate the Gini gain for each feature.

2.6 Support Vector Machine (SVM)

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed

as Support Vector Machine. Consider the below figure 2.6 in which there are two different categories that are classified using a decision boundary or hyperplane.

Example: SVM can be understood with the example that we have used in the KNN classifier. Suppose we see a strange cat that also has some features of dogs, so if we want a model that can accurately identify whether it is a cat or dog, so such a model can be created by using the SVM algorithm. We will first train our model with lots of images of cats and dogs so that it can learn about different features of

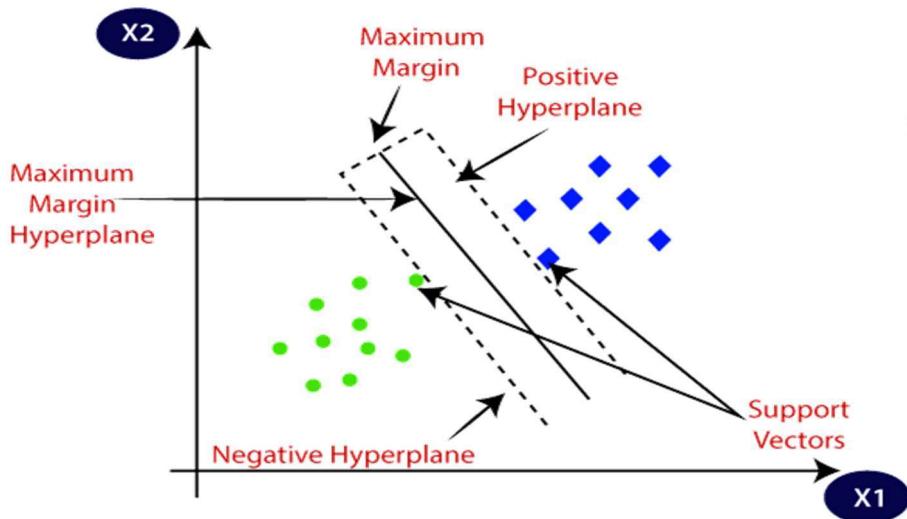


Figure 2.6: Hyperplane used in SVM

cats and dogs, and then we test it with this strange creature. So as support vector creates a decision boundary between these two data (cat and dog) and choose extreme cases (support vectors), it will see the extreme case of cat and dog. SVMs have several advantages, such as the ability to handle high-dimensional data and the ability to perform well with small datasets. They also have the ability to model non-linear decision boundaries, which can be very useful in many applications. However, SVMs can be sensitive to the choice of kernel, and they can be computationally expensive when the dataset is large. SVMs can also be used for regression tasks by allowing for some of the data points to be within the margin, rather than on the boundary. This allows for a more flexible boundary and can lead to better predictions. Based on the support vectors, it will classify it as a cat. Consider the below figure 2.7:

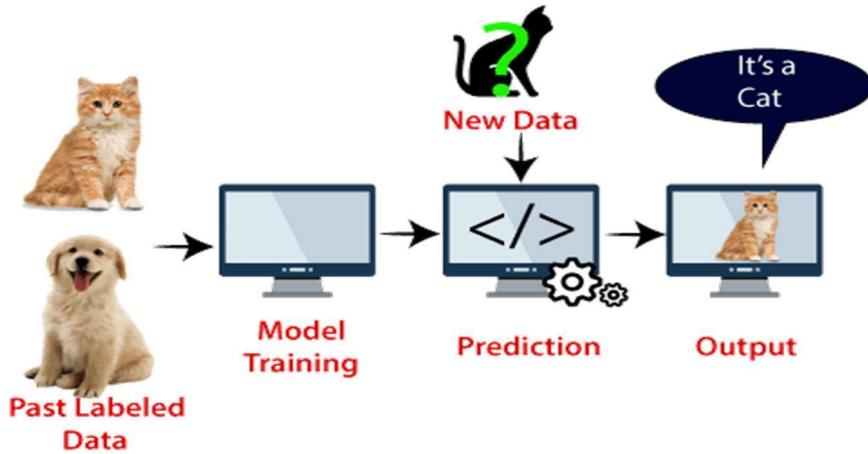


Figure 2.7: Working of SVM

SVM algorithm can be used for **Face detection, image classification, text categorization, etc.**

2.6.1 Types of SVM:

- **Linear SVM:** Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.
- **Non-linear SVM:** Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

2.6.2 Kernel Function:

Kernal Function is a method used to take data as input and transform it into the required form of processing data. “Kernel” is used due to a set of mathematical functions used in Support Vector Machine providing the window to manipulate the data. So, Kernel Function generally transforms the training set of data so that a non-linear decision surface is able to transform to a linear equation in a higher number of dimension spaces. Basically, It returns the inner product between two points in a standard feature dimension.

2.6.3 Major Kernel Functions:

For Implementing Kernel Functions, first of all, we have to install the “scikit-learn” library using the command prompt terminal: pip install scikit-learn

Linear Kernel: is used when the data is Linearly separable, that is, it can be separated using a single Line. It is one of the most common kernels to be used. It is mostly used when there are a Large number of Features in a particular Data Set. One of the examples where there are a lot of features, is Text Classification, as each alphabet is a new feature. So we mostly use Linear Kernel in Text Classification.

$$F(x) = B(0) + \sum(ai * (x,y)) \quad (5)$$

Sigmoid Kernel: This function is equivalent to a two-layer, perceptron model of the neural network, which is used as an activation function for artificial neurons shown in figure 2.8.

$$K(x,xi) = \tanh(\alpha xi \cdot xj + \beta) \quad (6)$$

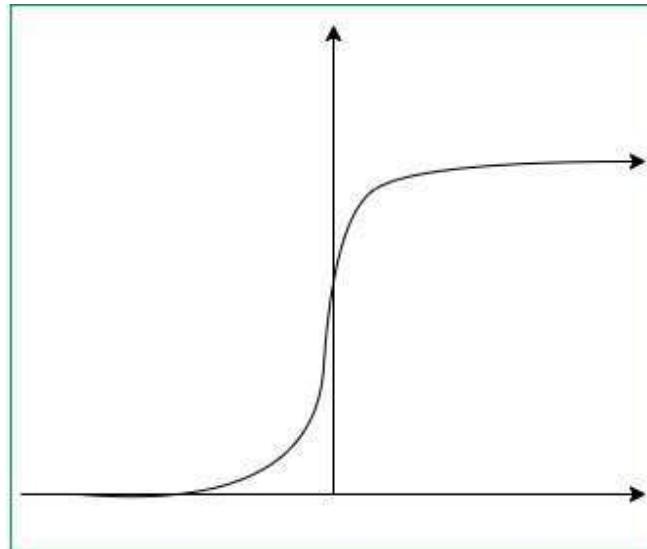


Figure 2.8:Sigmoid Kernel

Polynomial Kernel: It represents the similarity of vectors in the training set of data in a feature space over polynomials of the original variables used in the kernel and function is shown in figure 2.9.

$$K(x,xi) = 1 + \sum(x * xi)^d \quad (7)$$

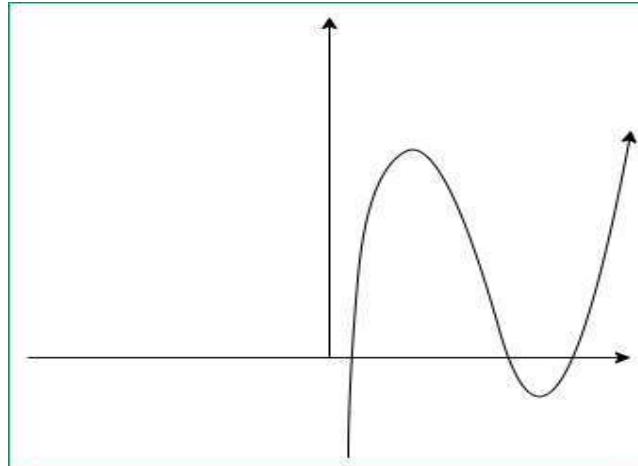


Figure 2.9: Polynomial Kernel

Gaussian Kernel Radial Basis Function (RBF): Same as above kernel function shown below in figure 2.10, adding radial basis method to improve the transformation.

$$K(x, x_i) = \exp(-\gamma \sum((x - x_i)^2)) \quad (8)$$

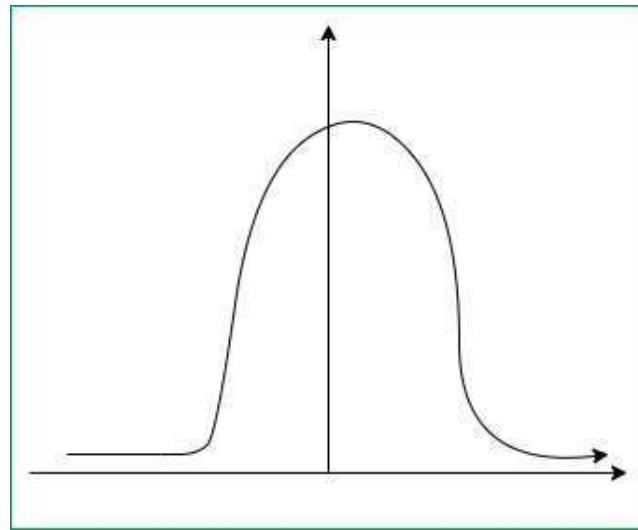


Figure 2.10: Gaussian Kernel

2.7 KNN Algorithm

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. K-NN algorithm assumes the similarity between the new case/data and

available cases and put the new case into the category that is most similar to the available categories. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suited category by using K- NN algorithm. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems. K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset. KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

Example: Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category as shown in figure 2.11.



Figure 2.11: KNN Classifier

2.7.1 Need of KNN:

Suppose there are two categories, i.e., Category A and Category B, and we have a new data point x_1 , so this data point will lie in which of these categories. To solve this type of problem,

we need a K-NN algorithm. With the help of K-NN, we can easily identify the category or class of a particular dataset. Consider the below figure 2.12:

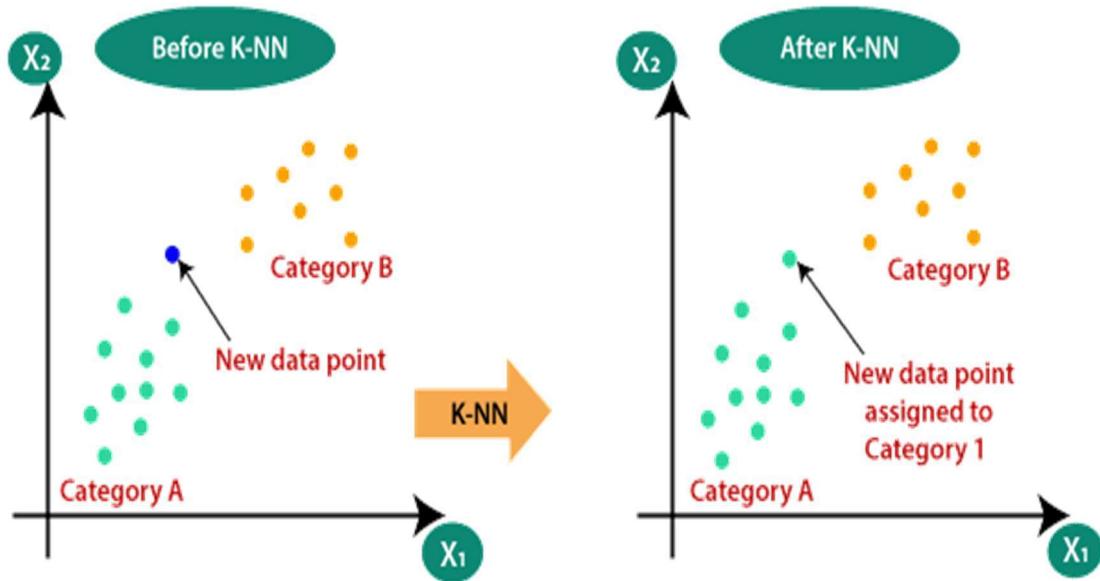


Figure 2.12: Category Graph

2.7.2 Working of KNN:

The K-NN working can be explained on the basis of the below algorithm and through figure 2.13:

- **Step-1:** Select the number K of the neighbours.
- **Step-2:** Calculate the Euclidean distance of K number of neighbours.
- **Step-3:** Take the K nearest neighbours as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbours, count the number of the data points in each category.
- **Step-5:** Assign the new data points to that category for which the number of the neighbour is maximum.
- **Step-6:** Our model is ready.

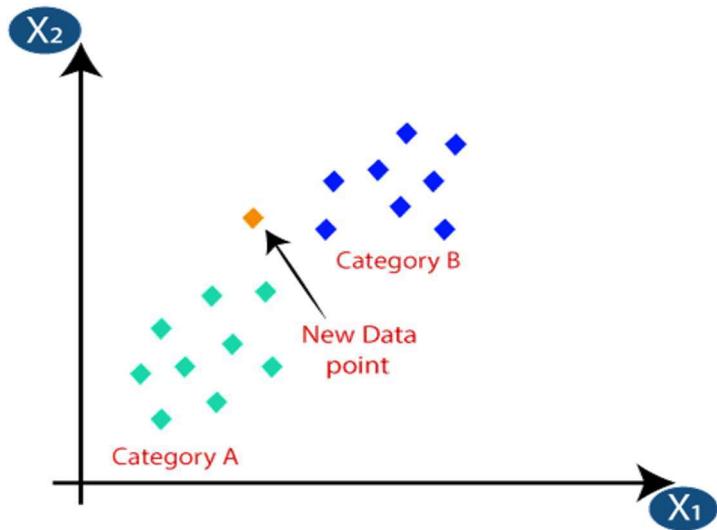


Figure 2.13: KNN Working

Next, we will calculate the Euclidean distance between the data points also shown in figure 2.14. The Euclidean distance is the distance between two points, which we have already studied in geometry. It can be calculated as:

$$\text{Euclidean Distance Between A and B} = \sqrt{(x_2^2 - x_1^2)^2 + (y_2^2 - y_1^2)^2} \quad (9)$$

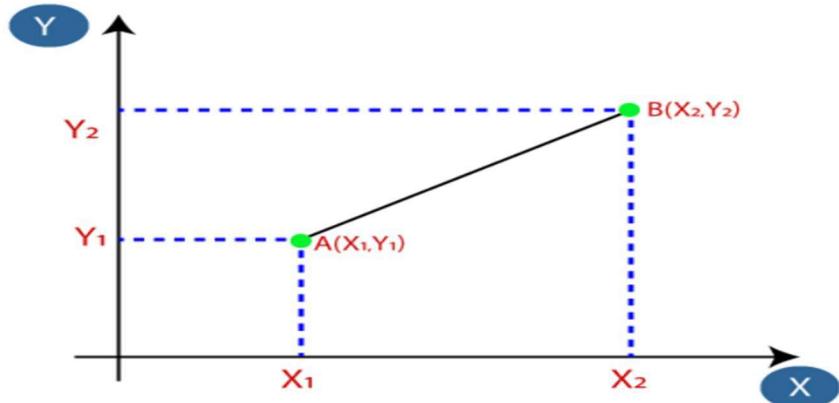


Figure 2.14: Euclidean Distance

By calculating the Euclidean distance we got the nearest neighbours, as three nearest neighbours in category A and two nearest neighbours in category B. Consider the below figure 2.15:



Figure 2.15: KNN Graph

As we can see the 3 nearest neighbours are from category A, hence this new data point must belong to category A.

2.7.3 Selection of K in the K-NN Algorithm

Below are some points to remember while selecting the value of K in the K-NN algorithm:

- There is no particular way to determine the best value for "K", so we need to try some values to find the best out of them. The most preferred value for K is 5.
- A very low value for K such as $K=1$ or $K=2$, can be noisy and lead to the effects of outliers in the model.
- Large values for K are good, but it may find some difficulties.

2.8 Random Forest Algorithm:

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting. The below figure 2.16 explains the working of the Random Forest algorithm:

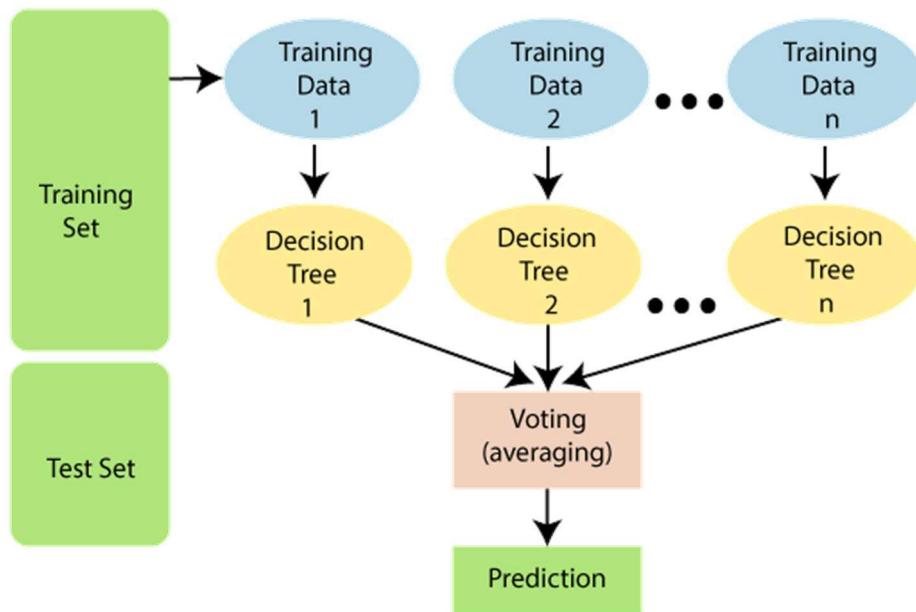


Figure 2.16: Random Forest Algorithm

Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. But together, all the trees predict the correct output. Therefore, below are two assumptions for a better Random

forest classifier. There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result. The predictions from each tree must have very low correlations.

2.8.1 Working:

The Working process can be explained in the below steps and diagram:

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

2.8.2 Advantages of Random Forest:

- Random Forest can perform both Classification and Regression tasks.
- It is capable of handling large datasets with high dimensionality.
- It enhances the accuracy of the model and prevents the overfitting issue.

2.9 Gradient Boosting Machine:

Machine learning is one of the most popular technologies to build predictive models for various complex regression and classification tasks. Gradient Boosting Machine (GBM) is considered one of the most powerful boosting algorithms.

Although, there are so many algorithms used in machine learning, boosting algorithms has become mainstream in the machine learning community across the world. Boosting technique follows the concept of ensemble learning, and hence it combines multiple simple models (weak learners or base estimators) to generate the final output. GBM is also used as an ensemble method in machine learning which converts the weak learners into strong learners. In this topic, "GBM in Machine Learning" we will discuss gradient machine learning algorithms, various boosting algorithms in machine learning, the history of GBM, how it works, various

terminologies used in GBM, etc. But before starting, first, understand the boosting concept and various boosting algorithms in machine learning.

Boosting is one of the popular learning ensemble modelling techniques used to build strong classifiers from various weak classifiers. In this way, this process of introducing more models is continued until we get a complete training data set by which model predicts correctly. AdaBoost (Adaptive boosting) was the first boosting algorithm to combine various weak classifiers into a single strong classifier in the history of machine learning. It primarily focuses to solve classification tasks such as binary classification.

There are a few important steps in boosting the algorithm as follows:

- Consider a dataset having different data points and initialize it.
- Now, give equal weight to each of the data points.
- Assume this weight as an input for the model.
- Identify the data points that are incorrectly classified.
- Increase the weight for data points in step 4.
- If you get appropriate output then terminate this process else follow steps 2 and 3 again.

2.9.1 Working:

Generally, most supervised learning algorithms are based on a single predictive model such as linear regression, penalized regression model, decision trees, etc. But there are some supervised algorithms in ML that depend on a combination of various models together through the ensemble. In other words, when multiple base models contribute their predictions, an average of all predictions is adapted by boosting algorithms.

Gradient boosting machines consist of 3 elements as follows:

- Loss function
- Weak learners
- Additive model

2.9.2 Loss function:

Although, there is a big family of Loss functions in machine learning that can be used depending on the type of tasks being solved. The use of the loss function is estimated by the demand of specific characteristics of the conditional distribution such as robustness. While using a loss function in our task, we must specify the loss function and the function to calculate the corresponding negative gradient. Once, we get these two functions, they can be implemented into gradient boosting machines easily. However, there are several loss functions have been already proposed for GBM algorithms.

Based on the type of response variable y , loss function can be classified into different types as follows:

2.9.2.1 Continuous response, $y \in \mathbb{R}$:

- Gaussian L2 loss function
- Laplace L1 loss function
- Huber loss function, δ specified
- Quantile loss function, α specified

2.9.2.2 Categorical response, $y \in \{0, 1\}$:

- Binomial loss function
- Adaboost loss function

2.9.2.3 Other families of response variables:

- Loss functions for survival models
- Loss functions count data
- Custom loss functions

2.9.3 Weak Learner:

Weak learners are the base learner models that learn from past errors and help in building a strong predictive model design for boosting algorithms in machine learning. Generally, decision trees work as a weak learner in boosting algorithms. Boosting is defined as the framework that continuously works to improve the output from base models. Many gradient

boosting applications allow you to "plugin" various classes of weak learners at your disposal. Hence, decision trees are most often used for weak (base) learners.

Training of weak learners:

Machine learning uses training datasets to train base learners and based on the prediction from the previous learner, it improves the performance by focusing on the rows of the training data where the previous tree had the largest errors or residuals. E.g. shallow trees are considered weak learner to decision trees as it contains a few splits. Generally, in boosting algorithms, trees having up to 6 splits are most common. Below is a sequence of training the weak learner to improve their performance where each tree is in the sequence with the previous tree's residuals. Further, we are introducing each new tree so that it can learn from the previous tree's errors. These are as follows:

1. Consider a data set and fit a decision tree into it.

$$F_1(x)=y \quad (10)$$

2. Fit the next decision tree with the largest errors of the previous tree.

$$h_1(x)=y? F_1(x) \quad (11)$$

3. Add this new tree to the algorithm by adding both in steps 1 and 2.

$$F_2(x)=F_1(x)+h_1(x) \quad (12)$$

4. Again fit the next decision tree with the residuals of the previous tree.

$$h_2(x)=y? F_2(x) \quad (13)$$

5. Repeat the same which we have done in step 3.

$$F_3(x)=F_2(x)+h_2(x) \quad (14)$$

Continue this process until some mechanism (i.e. cross-validation) tells us to stop. The final model here is a stagewise additive model of b individual trees:

$$f(x)=B \sum_{b=1}^B f_b(x) \quad (15)$$

Hence, trees are constructed greedily, choosing the best split points based on purity scores like Gini or minimizing the loss.

2.9.4 Additive Model:

The additive model is defined as adding trees to the model. Further, we can also prefer the gradient descent method by adding trees to reduce the loss. In the past few years, the gradient descent method was used to minimize the set of parameters such as the coefficient of the regression equation and weight in a neural network. After calculating error or loss, the weight parameter is used to minimize the error. But recently, most ML experts prefer weak learner sub-models or decision trees as a substitute for these parameters. In which, we have to add a tree in the model to reduce the error and improve the performance of that model. In this way, the prediction from the newly added tree is combined with the prediction from the existing series of trees to get a final prediction. This process continues until the loss reaches an acceptable level or is no longer improvement required.

CHAPTER 3

PROPOSED METHODOLOGY

3.1 Data Gathering

To evaluate the better result the quality and quantity of dataset is most important. Research got success only because the three different datasets and the tweets are of different types. This Twitter sentiment Analysis Datasets are taken from Kaggle Website ranging from 20000 to 1.6 Million of rows and 6 columns (target, ids, date, flag, user, tweet). Sample of dataset give in this Figure 3.1.

target	ids	date	flag	user	text
0	0 1467810369	Mon Apr 06 22:19:45 PDT 2009	NO_QUERY	_TheSpecialOne_	@switchfoot http://twitpic.com/2y1zl - Awww, t...
1	0 1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton	is upset that he can't update his Facebook by ...
2	0 1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus	@Kenichan I dived many times for the ball. Man...
3	0 1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF	my whole body feels itchy and like its on fire
4	0 1467811193	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	Karoli	@nationwideclass no, it's not behaving at all....

Figure 3.1: Sample of Data

Attributes of Dataset:

1. Target
 - a. Negative Tweets
 - b. Positive Tweets
 - c. Neutral Tweets
2. Ids
3. Date
4. Flag
5. User
6. Tweet

Graphical representation of Dataset:

The largest dataset of 1.6 million tweets is visualised in figure 3.2.

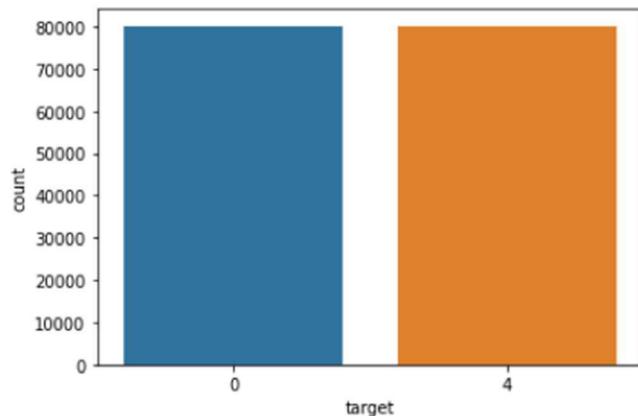


Figure 3.2: Graphical Representation of Data

3.2 Dataset without removing punctuation:

This section contains the dataset after removed punctuation as shown below in figure 3.3.

target	ids	date	flag	user	text
0	0 1467810369	Mon Apr 06 22:19:45 PDT 2009	NO_QUERY	_TheSpecialOne_	@switchfoot http://twitpic.com/2y1zl - Awww, t...
1	0 1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton	is upset that he can't update his Facebook by ...
2	0 1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus	@Kenichan I dived many times for the ball. Man...
3	0 1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF	my whole body feels itchy and like its on fire
4	0 1467811193	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	Karoli	@nationwideclass no, it's not behaving at all....

Figure 3.3: Dataset Before Clearing

3.3 Natural language processing (NLP):

Natural language processing is a subfield linguistic; computer science, information engineering, and artificial intelligence describes the interaction between human language and computers, in particular to program computers to process and analyse large amount of natural language data. Examples of NLP that we use in our everyday life:

- Spell check

- Autocomplete
- Spam filter
- Voice text messaging
- Siri, Alexa, or Google search engines

3.4 Pre processing:

Once the data is collected from the twitter the next step is pre-processing that is implemented in python. There are several steps involved in the pre-processing stage. They are,

1. Converting all uppercase letters to lowercase.
2. Tokenization generally done by installing the NLP package. It generally means removal of hash tags, numbers (1, 2, 3 etc.,), URL's and targets (@). Once tokenization is over we move to the next step of pre-processing.
3. Removal of non-English words Twitter generally supports more than 60 languages. But our project mainly involves English tweets; hence we remove the non-English words.
4. Emoticon replacements Emoticons are very important in determining the sentiment. Soothe emoticons are replaced by their polarity by seeing the emoticon dictionary.
5. Removal of stop words Stop words play a negative role in sentimental analysis, so it is important to be removed. They occur both in negative and positive tweets. A list of stop words like he, she, at, on, a, the, etc. are created and ignored. Once the above four steps are over we move to the next main method called feature extraction.

3.5 Dataset after cleaning:

For cleaning the dataset, we apply following methods:

1. Removing all the punctuations (letters expect [a-zA-Z])
2. Removing all the Stopwords(these are the words we don't want to use in tweets after cleaning the text which are not relevant to predict the sentiments are Positive, Negative and Neutral. These words may be like 'The', 'and', 'is', 'are' etc. These are the

word which did not give any hint about the sentiments is Positive, Negative and Neutral.

3. Make all the words of text in lower case.
4. Apply Stemming.

After Cleaning of data the below figure 3.4 shows the clear data.

```
print('Before: {}'.format(list(df['text'])[:2]))
print('---')
print('After: {}'.format(list(data[:2])))

Before: ['is drown in naked in revenge ', '@jjbalishhh i wanted to wait until you were
---
After: ['drown naked revenge', 'jjbalishhh wanted wait asleep could wake']
```



Figure 3.4: Dataset After Cleaning

3.6 Model Training:

Model training is a process of making an algorithm learn what it must do further. A Machine Learning model is train over a pre-defined dataset in which the output of the task is defined. The model training is performed by three types:

1. Supervised Learning
2. Unsupervised Learning
3. Reinforcement Learning

To make the validation set, there are two main options:

- Split the training set into two parts (60%, 20%) with a ratio 2:8 where each part contains an equal distribution of example types. We train the classifier with the largest part and make prediction with the smaller one to validate the model. This technique works well but has the disadvantage of our classifier not getting trained and validated on all examples in the data set (without counting the test set).

- The K-fold cross validation. We split the data set into k parts, hold out one, combine the others and train on them, then validate against the held-out portion. We repeat that process k times (each fold), holding out a different portion each time. Then we average the score measured for each fold to get a more accurate estimation of our model's performance.

These are the following models which are used in this project:

- Logistic Regression
- Support Vector Machine
 - SVM on Sigmoid Kernel
 - SVM on Radial Basis Function
 - SVM on Linear Kernel
 - SVM on Polynomial Kernel
- Decision Tree Classifier
- Random Forest Algorithm
- KNN Algorithm
- Gradient Boosting Machine

CHAPTER 4

RESULTS AND DISCUSSION

The different results of the variety of the algorithms are elaborated through tables and graphs. Table 4.1 shows the precision, recall and f1-score of all the category i.e., negative, neutral, and positive on each classifier used for the comparison. This helps in the analysis of the using logistical regression, naïve bayes, support vector machine with different kernels (i.e., sigmoid, polynomial, radial basis function and linear kernel), decision tree , random forest, knn, and gradient boosting machine.

After the prediction made by the different machine learning model, we can analyse the performance of each of the classifiers used in the machine learning model. From the table 4.2 we can get an idea of the performance of the classifiers. Figure 4.2 shows the confusion matrix of the Logistic regression model for an analysis of the true prediction as target is labeled on the y-axis and the predicted value is led on the x- axis. The confusion matrix is used to get the performance of the algorithm.

In Figure 4.2 the number of tweets whose sentiments are predicted correctly are placed in the diagonal of the matrix. The 0, 1 and 2 in the axis denotes the Negative, Neutral and Positive tweets classification. To compare between different algorithms, we have three parameters here precision, recall and f1- score. The mathematical formula for calculating each are as follows:

$$precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (16)$$

$$recall = \frac{True\ Positive}{True\ Positive + Fa\ Negative} \quad (17)$$

$$f1 - score = 2 * \frac{precision * recall}{precision + recall} \quad (18)$$

So, from the confusion matrix we can analyse that the sum of correct predictions is 287808 (72214 negative, 90262 neutral and 120788 positive) out of 320000 tweets. This gives an accuracy of 89.94% which is highest, among others. The other parameters are also derived

from the confusion matrix. Similarly, the Figure C shows the confusion matrix of the Decision Tree algorithm in which the correctly predicted tweets are 283264 out of 320000 tweets with an accuracy of 88.52%. On concluding, we can see that the highest accuracy is obtained from the Logistic Regression algorithm followed by Decision Tree Algorithm, Random Forest Algorithm, Support Vector Machine, Gradient Boost Algorithm and KNN in order from highest to lowest. Figure 4.1 shows a plot of the accuracy of different algorithms on a linear graph for a visualized idea of the performance of each classifier.

Table 4.1: Performance Parameter

Algorithms		Positive			Negative			Neutral		
		Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Support Vector Machine	Linear	0.92	0.91	0.87	0.87	0.84	0.89	0.91	0.87	0.89
	Polynomial	0.89	0.88	0.94	0.88	0.84	0.89	0.91	0.88	0.94
	Sigmoid	0.91	0.84	0.76	0.85	0.88	0.87	0.86	0.81	0.84
	RBF	0.93	0.87	0.87	0.88	0.84	0.91	0.88	0.87	0.83
Decision Tree		0.88	0.99	0.93	0.87	0.81	0.84	0.90	0.83	0.86
Random Forest		0.90	0.86	0.88	0.83	0.69	0.75	0.83	0.96	0.89
KNN		0.92	0.25	0.40	0.74	0.18	0.28	0.42	0.99	0.59
Gradient Boost Machine		0.92	0.86	0.75	0.68	0.95	0.75	0.91	0.89	0.89
Logistic Regression		0.94	0.90	0.92	0.89	0.75	0.82	0.86	0.98	0.92

Table 4.2: Accuracy Comparison

No.	Algorithms	Accuracy
1	Logistic Regression	89.94
2	Support Vector Machine	78.65
3		83.84
4		81.59
5		81.86
6	Decision Tree	88.52
7	Random Forest	85.93
8	KNN	49.73
9	Gradient Boost Machine	78.68

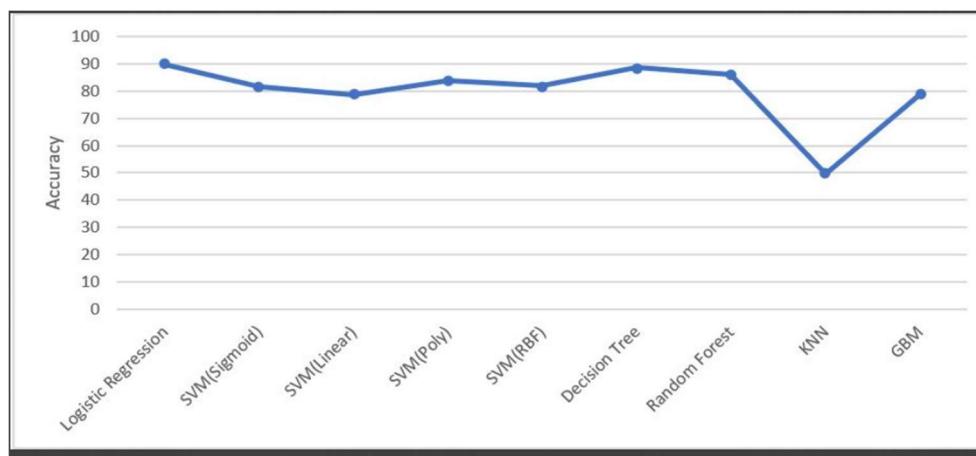


Figure 4.1: Accuracy Graph

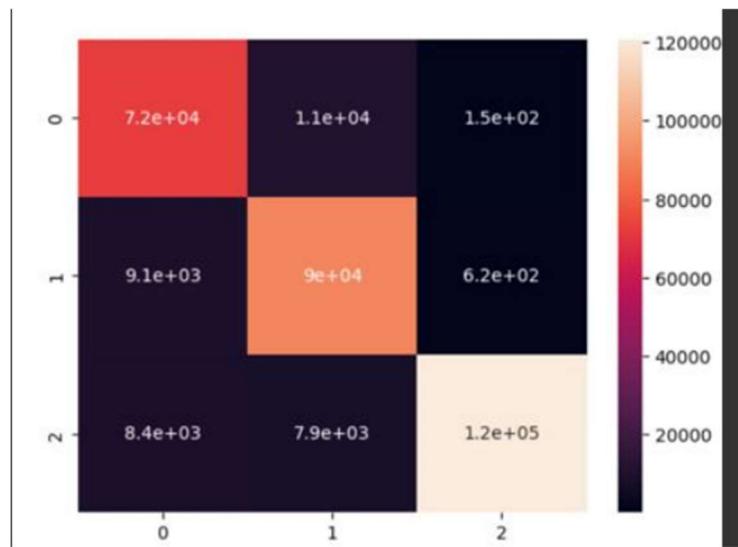


Figure 4.2: Confusion Matrix

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

In this project the overall analysis of different models based on machine learning are used to analyse the sentiments. The analyses done on the data set which we have trained in this project provide us the following result:

The best model for sentiment analyses is Logistic Regression for our dataset which provides an accuracy of 89.94%. The second model is Decision Tree Algorithms which provides an accuracy of 88.52%. But this accuracy can change depending on the different dataset. The result of accuracy will vary according to the type of dataset. Figure 1 shows a plot of the accuracy of different algorithms on a linear graph. It shows the performance of different models on the tweets.

5.2 Future Scope

Tweeter is a famous platform for opinion sharing about any topic, product or anything going on in mass. So, to get an idea about anything going in the public people uses twitter and read the opinion about that topic. In recent years, most of the public uses twitter for sharing their review and opinion over any protest going on, product launched by any company or any public problem so that their opinion can reach in mass. The analysis we have done here classifies some of the best algorithms sentiment analysis of any textual opinion.

The model trained here can be used for the analysis of the sentiment of the public or mass about any product in commercial field so that the company can extract the negative opinion among all the opinions and enhance the product. This will give a boost in the marketing of the product and helps the company to find out the negativity in the product. In commercial area, when a company wants to introduce any new product, he can analyse the feedback of the similar past product to take a brief idea of the opinion of the public.

Similarly, in political context by designing a UI a person can get the opinion about any one or any party to analyse their image in public. One drawback of our algorithm is that we have used classifiers for the analysis of sentiments. The improvement can be done by using reinforcement learning algorithms by implementing TensorFlow library from python. TensorFlow uses perceptron model to predict the result giving a better accuracy with a less training time needed.

REFERENCES

- [1] Prajval Sudhir, Varun Deshakulkarni Suresh, “Comparative Study of Various Approaches, Application and Classifiers for Sentiment Analysis”, in Global Transitions Proceedings, 2021.
- [2] Ashwin Sanjay Neogi, Kirti Anilkumar Garg, Ram Krishn Mishra, Yogesh K Dwivedi, “Sentiment Analysis and Classification of Indian Protest Using Twitter Data”, in International Journal of Information Management Data Insights, 2021.
- [3] Anupama B S, Rakshith D B, Rahul Kumar M, Navaneeth M, “Real Time Twitter Sentiment Analysis using Natural Language Processing”, in International Journal of Engineering Research and Technology, Vol.09, Isuuue 07, ISSN:2278-0181, 2020.
- [4] Saurabh Singh, “Twitter Sentiments Analysis Using Machine Learning”, in International Journal of Scientific Research in Computer Science, Engineering and Information Technology, ISSN:2456-3307, 2020. [5] Faizan, “Twitter Sentiment Analysis”, in International Journal of Innovative Science and Research Technology, ISSN No:2456-2165, 2019.
- [6] Vishal Jain, Mahesh Parmar, “A Review on Emotion and Sentiment analysis Using Learning Techniques”, in International Journal for Research in Applied Science & Engineering Technology, Vol 10, ISSN: 2321-9653, 2022.
- [7] Ayushi Mitra, “Sentiment Analysis Using Machine Learning Approaches (Lexicon based on movie review dataset)”, in Journal of Ubiquitous Computing and Communication Technology, Vol.02/ No.03, pp 145-152, 2020.
- [8] Ruijun Liu, Yuqian Shi, Changjiang Ji, Ming Jia, “A Survey of Sentiment Analysis Based on Transfer Learning”, supported by National Natural Science Foundation of China, 2019.
- [9] Laszlo Nemes, Attila Kiss, “Social media sentiment analysis based on COVID-19”, in Journal of Information and Telecommunication, DOI: 10.1080/24751839.2020.
- [10] Huyen Trang Phan, Van Cuong Tran, Ngoc Thanh Nguyen, Dosam Hwang, “Improving the Performance of Sentiment Analysis of Tweets Containing Fuzzy Sentiment Using the Feature

Ensemble Model”, supported by Basic Science Research Program through National Research Foundation of Korea, 2020.

[11] Mickel Hoang, Oskar Alija Bihorac, Jacobo Rouces, “Aspect-Based Sentiment Analysis Using BERT”.

[12] Ashima Yadav, Dinesh Kumar Vishwakarma “Sentiment analysis using deep learning architectures: a review”, Springer Nature B.V. 2019.

[13] Abdullah Alsaeedi, Mohammad Zubair Khan, “A Study on Sentiment Analysis Techniques of Twitter Data”, in International Journal of Advanced Computer Science and Application, Vol.10/ No.02, 2019.

[14] A.M. Johm-Otumu, M. M. Rahman, O. C. Nwokonkwo, M. C. Onuoha “AI-Based Techniques for Online Social Media Network Sentiment Analysis: A Methodical Review”, in International Journal of Computer and Information Engineering, Vol.16, No.12, 2022.

[15] Hao Tian, Can Gao, Xinyan Xiao, Hao Liu, Bolei He, Hua Wu, Haifeng Wang, Feng Wu, “SKEP: Sentiment Knowledge Enhanced Pre-training for Sentiment Analysis”, in arXiv:2005.05635v2, 2020.

[16] Samira Zad, Maryam Heidari, James H Jr Jones, Ozlem Uzuner, “A Survey on Concept-Level Sentiment Analysis Techniques of Textual Data”, in IEEE World AI IoT Congress, 2021.

[17] Kamaran H. Manguri, Rebaz N. Ramadhan, Pshko R. Mohammed Amin, “Twitter Sentiment Analysis on Worldwide Covid-19 Outbreaks”, in Kurdistan Journal of Applied Research (KJAR), ISSN:2411-7706 2020.

[18] Koena Ronny Mabokela, Turgay Celik, Mpho Raborife “Multilingual Sentiment Analysis for Under-Resourced Languages: A Systematic Review of the Landscape”, supported by National Research Foundation for the Black Academics Advancement Programme.

[19] Li Yang, Ying Li, Jin Wang, R. Simon Sherratt, “Sentiment Analysis for E-Commerce Product Reviews in Chinese Based on Sentiment Lexicon and Deep Learning”, supported in part by the National Natural Science Foundation of China, 2020.

- [20] Vishal A Kharde, S.S. Sonawane, “Sentiment Analysis of Twitter Data: A Survey of Techniques”, in International Journal of Computer Application, 2016.

RESEARCH PAPER

A Machine Learning Approach of Sentiment Analysis on Twitter

Dr. Sanjiv Sharma, Mirza Tabish Hassan, Apurva Goel, Nazakat Ali Sofi

Department of Computer Science and Engineering, KIET Group of Institutions, Ghaziabad, UP, India

Abstract: With the recent growth of social media like social networks, internet sites, and online blogging, users share a lot of opinions, reviews, ratings, and feedback on these platforms. This information is very useful for businesses, governments, and people. From these reviews, comments, opinions and rating, the mood or sentiment of users can be inferred. The sentiment analysis is a process of identification of the sentiment of textual data using some algorithm or tool. Twitter is a globally accepted micro-blogging site developed for opinion sharing in form of tweets. This makes tweets perfect as a dataset to analyze sentiment. This paper proposed a framework for the sentiment analysis using machine learning algorithms. To validate the proposed framework this work uses three different datasets of tweets from online data repository Kaggle, a natural language tool kit algorithm TF-IDF, is used to convert text to vectorized tweet, and six machine learning algorithms. A comparative analysis is also done on the basis of the accuracy along with performance parameter like precision, f1-score and recall of each of the machine learning models.

Keywords: Machine Learning, weblog, Sentiment Analysis, TF-IDF

1. Introduction

Online social media platforms are the way to share the opinions, thoughts, suggestions about any topic or product by the public. These platforms are used by different communities like political, commercial, or social to share their ideas, work to the public for the promotion. The social media platforms are also used by different commercial companies to advertise their product, service and share their benefits to the public. The political sector also uses different social media platforms to promote their work in the public and get recognition. The public uses these platforms to share their feedback about any political work, commercial product or social issue. The feedback given by the public is used to enhance the product or service of any company. It is also used to develop new business strategies as per the public demand.

According to Drus et al. , [1] there are different social media platforms which is used to share the opinions, thoughts, and emotions of the public like Facebook, Instagram, Twitter etc. The Facebook and Instagram are also used for sharing the status, picture of the someone's life along with their opinions while Twitter is a social media platform which is mainly used for the purpose sharing the opinions, thoughts, emotions and feedback. A proper analysis of content available on social media platforms may be beneficial for any organization or government. As per study by Kharde et al. , [2] the process of Sentiment Analysis is a technique of analyzing the tweets and determining the sentiment in the form of positive, neutral, or negative remarks. Agarwal et al. studied that [3] studied that these categories of the tweets are classified on the basis of their polarity scaling from -1 to 1. The tweets having the polarity range from -1 to less than 0 are classified as negative tweets. The tweets having polarity scale equal to 0

are classified as neutral tweets. The rest of tweets having polarity ranging from greater than 0 to 1 is classified under positive tweets. This study provides a way of extracting sentiment of mass from the available tweets on Twitter platform. In order to analyze the sentiment, this paper uses the three different datasets from the reputed online repository Kaggle. After getting the tweets, six supervised machine learning algorithms are used to classify the sentiments of the tweets gathered. Rest of the paper is divided in nine sections. Section 1 describes the brief introduction about the paper. Section 2 shows the study of related work which is available on the online repository. Section 3 describes the flow of the methods used in the work done for retrieving the conclusion of the paper. Section 4 describes about the source from which we have gathered the data. Section 5 refers to a detailed idea of the pre-processing of dataset. Section 6 refers to the detailed study of the different machine learning algorithms used in the paper. Section 7 is the result and discussion which refers to the analysis of the performance parameter and the result gathered from different algorithms of machine learning. Section 8 is the conclusion which is derived from this paper. Section 9 describes the future use of the research in detail.

In the literature review, this study has identified that the dataset frequently used for analysis is very small consisting of tweets in a range of 17000-25000. Due to a small data set, the machine learning algorithm is trained over a small number of features. This led to the low accuracy and the performance of the model. The model also lacks a huge number of features which can result in the false prediction of sentiments of any tweets consisting of new features. To overcome this problem, proposed work use multiple Kaggle datasets containing tweets ranging from 20000 to 1600000. Following are the contribution of this research work:

1. A relatively large and latest dataset containing 1600000 tweets are used in this work to improve the performance of machine learning model and get a real insight.
2. Using multiple dataset containing a wide range of tweets (dataset 1 contains 20000 tweets, dataset contains 27481 tweets and dataset 3 contains 1600000 tweets), we have done a comparative study that how different algorithms performs on different dataset having different number of features.
3. Six supervised machine learning algorithms are used for a brief analysis of the performance of these algorithms on the problem of sentiment analysis.

2. Related Works

Sudhir et al. , [4] describes about the different processes used for the sentiment analysis. The paper mentions the different approaches of machine learning which can be used for the implementation of sentiment analysis. The techniques the paper used are the lexicon-based sentiment analysis, machine learning approach and rule-based approach. The classification algorithms the paper had described are deep learning techniques. These techniques include bidirectional encoder representations from transformers (BERT), LSTM (Long Short-Term Memory Model) and GRU (Gated Recurrent Unit). The paper has also discussed about some of the machine learning algorithms that include Decision Tree, K-NN, Naive Bayes, SVM. The paper does a comparative study between all these algorithms and classifiers. A comparison of these traditional algorithm is also done in a paper written by Dhola et al. , [5] The papers have taken different performance parameters like precision, f1-score, recall and accuracy to analyze the best algorithm and classifier for sentiment analysis.

Neogi et al. , [6] had discussed about the sentiment analysis of tweets fetched from the twitter api using different machine learning algorithm. The aim of the research was to survey and inspect the feelings and emotions of public especially farmer community about their objection and opposition. In this research the data is gathered from social media platform, Twitter using twitter api. On daily basis, 200 tweets were fetched from twitter using hashtag like '#FarmerProtest'. The data collected from the twitter is appended in a list data structure in python for a duration of 5 months. The paper uses a set of 18000 tweets as a dataset. After data gathering, the pre-processing of the tweets is done to make the tweets free from unwanted tokens. The dataset is passed through the TextBlob to analyze the sentiments for supervised learning of algorithms used in the research. The vectorization of the data is done through 2 methods for fitting the data in machine learning algorithm. The paper uses both bag of words and TF-IDF for this process. After the tweets have been converted into the tokens, the research uses 4 machine learning algorithms for prediction of the sentiments. The dataset is splitted in 2:8 ratio for testing and training set respectively. The training set is used to train the algorithm like naïve bayes, support vector machine, decision tree classifier and random forest. The paper concludes that the support vector machine with bag of word vectorization has the highest accuracy.

According to the study [7] which is the analysis of the sentiments of any real time tweet using a website as interface. The paper uses natural language processing for the cleaning of the data. Naïve Bayes is used for the prediction of the sentiment of the tweet fetched from the twitter using api. The algorithm is trained using training dataset and integrated with the user interface. The training dataset used in the paper is a movie review from imdb. The user interface accepts the user handle, tweets of any user and fetch an appropriate number of tweets from twitter. After the tweets has been stored into a data frame, pre trained naïve bayes machine learning algorithm is used to predict the sentiment of the tweets. The classification of the tweets is plot on a graph using matplotlib. The graph is displayed on the website. Along with the graph, the user can see the positive, negative and neutral tweets separately. The result of the paper was a real time sentiment analysis on tweets fetched from the api of twitter on any specific topic, user, or type.

The author of the paper Singh [8] has analyzed the accuracy and performance parameter like precision, recall and f1-score of naïve bayes supervised machine learning algorithm. The data is gathered from an online repository having tweets with their sentiments as target. The cleaning of the tweets was done by using natural language processing. The paper uses multiple techniques for the cleaning of tweets under natural language processing like stemming, lexicon analysis and tokenization. After the data has been preprocessed, bag of word is used to represent the tweets into numerical form so that the machine can understand. The data is splitted into training and testing set in appropriate ratio. The machine learning algorithm is trained on the training set of data. Further, the model is used to predict the sentiment of the tweets in testing dataset. The result is compared with the original target and the performance parameter like accuracy and precision is calculated.

According to Faizan [9], it is shown that millions of social media users generating large amount of data and information (big data) that can both structured and unstructured. Every person influenced from social networking sites feels great to share their views and thinking about the present action or state they had gone through. This information is expressed to put the views of users about any chosen topic or issue. This research derives the feelings behind these posts and selected Twitter platform as the data source to begin the work. This research examines and inspect the methods of pre-processing and classification of textual data using python to find the sentiment as outcome of this textual twitter data. The information

and textual data are collected from the social networking site Twitter with the use of API of Twitter by using the access token and access token secret for the need of authorization and identification. After the data has been collected, TextBlob is used to compute the target value for training the algorithm. The target is computed in form of the polarity and subjectivity. On the basis of polarity and subjectivity, the tweets are classified in positive, negative and neutral. The NLTK is used for data processing and the machine learning technique k-nearest neighbor is used for the classification of data against the target. The classified tweets are compared with the target and performance parameter is computed.

Jain et al. , [10] aims to analyze the sentiments of the feedback of any product in various levels. The sentiments are classified into three category positive, negative and neutral having different levels of sentiments. The paper describes about different techniques of sentiment analysis and there use cases. According to Nadwani et al. , [11]The techniques used in the paper are deep learning, different machine learning algorithms and lexicon-based methods. The paper also discusses about the different problems faced during the analysis of sentiments using these techniques. The paper concluded that the deep learning method is an appropriate method for the classification of sentiments of any huge data. Along with this, the paper also concluded that if public gives the proper feedback of the merchandise or product, they buy from the market then sentiment analysis will help other to know about the cons and pros of the product.

Mitra [12] discuss about the techniques of machine learning for the analysis of sentiment of a movie review. The data set use for the analysis of the sentiments is a movie review taken from an online repository imdb. Further, natural language processing is used for the cleaning of the review. Using different techniques like stemming, tokenization, stop-word removal and advance python programming, the tokens (words or fragments of any textual data) which does not contribute to the analysis is removed. The pre-processed data is further passed through different algorithms of machine learning one by one. The machine learning algorithm used for analysis are naïve bayes, support vector machine, random forest, decision tree classifier and k-nearest neighbor. The performance parameter and accuracy are written down in a table and a comparative analysis is done in the research.

Liu et al. , [13] discuss about the classification of the sentiments in form of positive, negative and neutral by using transfer learning approach. The data is gathered form different online repository like imdb, Kaggle and other. The data used for the research is the product review on amazon and movie review form imdb. The pre-processing of the data is done by passing the data through different natural language processing techniques. The preprocessing removes the unwanted words including punctuations and numbers which does not affect the accuracy of the prediction. After the preprocessing, transfer learning mechanism of machine learning is used. In transfer learning, the algorithm is trained on a domain and then tested on different domain. In the paper, for fitting different domain in the model data has been fetched from 5 source like are IMDB, Stanford sentiment Treebank, YELP, SENTIMENT140 (STS) and Amazon product. The paper also discusses about the different techniques which can be used to analyze the sentiments of any text. The paper concluded that among different methodology AWD-LSTM has the highest accuracy of 95.40%.

Nemes et al. , [14] discuss about the sentiment analysis of data using recurrent neural network in deep learning. The data used for the analysis is related to the covid-19 and fetched from the twitter api. The data is fetched on daily basis over a month and appended in a data frame. The pre-processing of the data gathered is done using stemming and tokenization. After the data has been pre-processed, the sentiment is classified using TextBlob, which is inbuilt sentiment analyzing library in python. The

sentiment analyzed from the TextBlob is used as target for the machine learning algorithm. A recurrent neural network from deep learning is used as a machine learning model. The model is trained and tested over the data gathered. The paper gives the public sentiments about the covid-19 and how bad the pandemic hits the mass. The paper also mentions about a user interface for a real time analysis of sentiments using machine learning.

The paper proposed by Phan et al. , [15] discuss about the way of improving the efficiency using feature ensemble learning. The paper has proposed a method of sentiment analysis which can improve the efficiency of the algorithms. The method proposed in the paper is started by fetching the raw tweets using the twitter api or any online resource. The data is cleaned by removing the punctuations, URL, hashtag, and other tokens which does not contribute to the performance analysis. The data tokenization and POS-tagging is done after the cleaning for the pre-processing. After the data has been pre-processed, features of the preprocessed data are extracted like N-gram, Negative words, Positive words etc. The features extracted is then converted into feature vectors. This vector is passed through convolution artificial intelligence layer for training of the network. When the model is trained, the sentiment is predicted in 5 categories as strong positive, positive, neutral, negative and strong negative. This provides, a brief classification of the sentiments with their levels.

In the paper Hoang et al. , [16] discuss about the BERT i.e., bidirectional encoder representations from transformer for the analysis of sentiments. The convectional method uses unidirectional encoder for different nlp techniques like sentiment analysis. The paper discussed that the performance can be increased by using bidirectional encoder. The bidirectional encoder technique encodes the words form both directions i.e., right to left and left to right. The paper also used aspect learning method for the analysis of sentiment. In this method words having a sentiment in a sentence is related to the previous work for the analysis. The paper uses two levels of the aspect models i.e., sentence level and text level. The paper has analyzed the sentiment on both the levels and concluded that the performance of the natural language processing can be improve by using BERT with a dataset having more unique aspect.

Yadav et al. , [17] proposed different sentiment analysis techniques using deep learning. The different datasets mentioned in the paper are IMDB, Yelp, Amazon review dataset, MOUD dataset, Getty Images dataset, Twitter image dataset, CMU-MOSI dataset, and Stanford sentiment Treebank dataset. The data is cleaned and preprocessed using convectional method. The different deep learning techniques used in the research are Convolutional neural networks (CNNs), Recursive neural networks, long short-term memory, Recurrent neural networks, Gated recurrent units and Deep Belief Networks (DBNs). These algorithms are used to train different models and all the models are tested on testing set. The performance parameters are computed, and a comparative study of these model is done in the paper. The paper also discussed about the advantages and disadvantages of each algorithm and concluded that the model having CNN, LSTM, Attention, and late fusion has the highest accuracy of 96.40%.

Alsaedi et al. , [18] discuss about document level and sentence level sentiment analysis along with twitter sentiment analysis. In document level approach, document is given as an input and sentiment of each document is analyzed using machine learning. In sentence level approach, the sentiment of each sentence is analyzed using machine learning. The paper also discusses about the different classification techniques which can be implemented on these approached for prediction of sentiment. The classification techniques discussed are Naïve bayes, Support vector machine and maximum entropy method. The paper concludes that the support vector machine on data fetched from twitter api has an accuracy of 92%.

A. M. Jhon-Otumu et al. , [19] describes about the AI based sentiment analysis with an improved efficiency. The data used for the research is fetch from 3 online repository which are google scholar, ieee xplore and acm digital library. The data used in the research has 52 list (26 from google scholar, 25 from ieee xplore and 1 from acm digital library). After the data has been gathered, different classifiers with CNN are used for the model training and prediction. The model used in the paper are CNN+LSTM, CNN+BERT, CNN+RNN and CNN+PNN(BiGRU). The paper Aslan et al. , [20] also compute the sentiments using optimized CNN. The method followed by the research is started with the extraction of word features. After the word features are extracted, the data is splitted into training and testing set which is used for the training and prediction of the sentiments. The paper concludes that the model having CNN with LSTM has the highest accuracy of 92%.

According to Tian et al. , [21], in traditional sentiment analysis, the sentiment words and aspect sentiment pairs are considered while training the model. In new approach of pre training, these features are neglected[2] which can decrease the efficiency of the algorithm. The paper discusses about a technique known as sentiment knowledge enhance pre-training. In the paper sentence level approach is followed. The data is gathered from two online source that are Standford Sentiment Treebank and Amazon-2. The data is cleaned and preprocessed. After the data has been prepared for the model training, neural network is used as model. The model training uses around 10 epochs on the sst-2 dataset and 3 epochs on amazon dataset. The model is then evaluated on the testing set and the performance of each is compared.

The author Zad et al. , [22] discusses about the process of sentiment analysis on any textual data. In this paper the sentiment is analyzed on the concept level of the data. The textual data gathered from any source is passed through a cleaning process. The words, numbers and punctuations which does not contribute to the analysis is removed in cleaning. The cleaned data is then passed through a stemmer in which the words in the sentence are converted into their root form. The sentence is then tokenized, and the feature is extracted from the data. Based on the features extracted from the data the sentiment is determined. The paper discusses about some of the machine learning techniques such as naïve bayes, support vector machine, Bayesian network, decision tree classifier and neural network. These algorithms are used for determining the sentiments on different domains of data. The paper mentioned about social media, marketing and product review.

According to H. Manguri et al. , [23] and Dubey Assistant Professor n.d. , [24], during Covid 19 pandemic, people were started sharing their opinion on online platform as they were not able to meet other. The paper focused on the sentiment analysis of the tweets done by the people on twitter on covid 19. The data in the research is fetched using twitter api on daily basis and appended in a data frame. In a week, 530232 tweets have been gathered for the further process. After the data has been fetched, cleaning and tokenizing of the dataset is done. The tweets are categorized in three categories (positive, negative and neutral) with the help of TextBlob. After the classification, the data is splitted into training and testing sets for training and performance evaluation respectively. Naïve bayes algorithm is used as a model in the paper. The model is trained on training set under supervised learning algorithm and the testing set is used form evaluating the performance parameters.

The research done by Mabokela et al. , [25], discusses about the sentiment analysis of textual data in multiple language. The main focus of the research is determining the sentiments of under-resourced language. The under resourced language refers to the language having minimal resource for the digitalization. The research uses two methodology one is machine translation and other is machine learning for the conversion of the language into English. After the language has been translated in English,

the preprocessing of the data is done. After preprocessing, the sentiment is analyzed by using co-training model. In co-training model, the trading is continued while the sentiment is analyzed. The paper concluded that using co-training method for multilingual sentiment analysis can improve the accuracy of the model.

The paper proposed by Yang et al. , [26] discusses about the product review of any e-commerce product in Chinese. The model used for the product review in the research is deep learning. In the research, the advantages of sentiment lexicon, CNN model, GRU model and attention mechanism is combined to improve the accuracy. A sentiment lexicon is constructed which is used to give a weight corresponding to words. After the construction of sentiment lexicon, different layers are combined in series to improve the accuracy. The layers combined are embedded layer followed by convolution layer, pooling layer, BiGRU layer and attention layer. The model is trained on the dataset containing 100000 reviews fetched from Dangdang online repository. The data is pre-processed using natural language processing. After the preprocessing, the data is passed through the layers for training and testing. The performance parameters like precision, accuracy and f1-score are evaluated. The paper concluded that the accuracy of the model is 93.5% on 15 epochs on the model.

The work done by Kharde et al. , [27] discuss about the sentiment analysis of tweets fetched from the twitter using supervised machine learning algorithm. The data for the research is gathered from the publicly available online source and a total of 250000 tweets have been used here. The dataset has the sentiment column where sentiment of each tweet is already given. This is used for the supervised learning of an algorithm. The data is cleaned and tokenized using natural language processing and python programming. The features like words and their frequencies, parts of speech tags, opinion words and phrases and negations are extracted. Then the dataset is splitted into training and testing sets. The training set is used for the training of the machine learning algorithm and the testing set is used for comparing the predicted value with expected value. The machine learning algorithm used here are naïve bayes, maximum entropy and support vector machine. The paper concludes with a comparative result between different algorithms in which support vector machine has got the highest accuracy of 86.40%.

3. Taxonomy of research

This section of the paper represents the workflow of the process used in the research in form of a flow diagram. The research starts with the gathering of the multiple datasets from an online reputed repository Kaggle. Three datasets are used for a comparative study of six machine learning algorithm on different size of datasets. After that, the pre-processing of the data is done using NLP. There is various method of pre-processing of the data but the best among all is natural language processing which is being used in our research. The processed data is then splitted into two datasets (training set and testing set). Further a model of machine learning is selected. The model is trained over the training dataset. After training of the model, the testing dataset is provided to the model and performance parameter is evaluated. This step after the processing is repeated with multiple models of machine learning (in this paper 9 algorithms of machine learning is used). At the end a detailed comparative analysis of the results of different machine learning algorithm is done to obtain the conclusion. The workflow is visualized in the Figure 1.

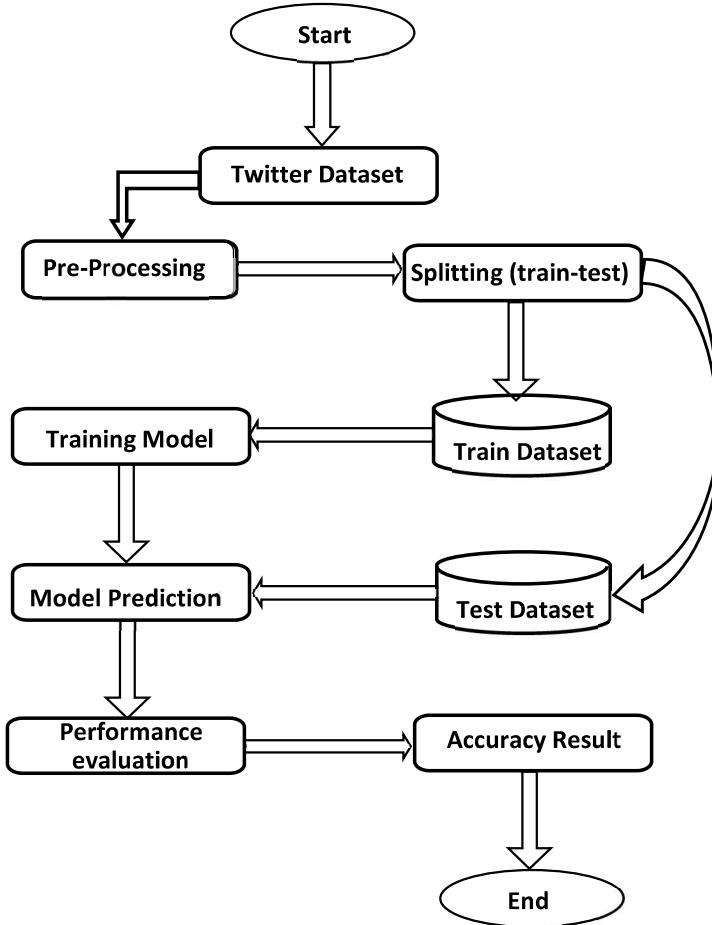


Fig. 1. Flow-Chart of the methodology used in the project.

Algorithm:

- 1: Initialize data frame df
- 2: Input:Raw CSV files of tweets with their sentiments in a data frame df
- 3: Initialize stopword set form nltk
- 4: for each tweet in df:
 - 5: remove web links and punctuations
 - 6: remove stopwords from tweets using stopword set
- 7: for each tweet in df:
 - 8: vectorize tweets using tf-idf
- 9: split vectorize tweets and sentiments from df in a ratio of 8:2 as train_x, test_x and train_y,test_y respectively
- 10: Import classifiers form sklearn
- 11: Train the classifiers using train_x and train_y
- 12: Initialize sentiment = predict(test_x)
- 13: compare sentiment with test_y and print the performance parameters.

4. Data Acquisition:

To evaluate the better results the quality and quantity of Dataset plays the key role and become one of the reasons for the success of research. The three datasets used in the work for the training and

testing purpose is retrieved from an online repository named Kaggle. All of the three datasets are being split in two different sets one for the training and other for the testing purpose. Each of the datasets are used for the testing and training purposes one after other and a brief comparative study is done further. The quantity of tweets with their sentiments of each dataset is listed in table 1.

Table 1 Datasets used in this work.

	Positive	Negative	Neutral	Total
[28]Dataset 1	10992	8708	7781	27481
[29]Dataset 2	5600	6000	8400	20000
[30]Dataset 3	656000	512000	432000	1600000

5. Pre-processing:

In pre-processing, we have used multiple techniques to remove the unwanted words and make all the words to reflect to their root form. The process of preparation of dataset for the analysis is known as pre-processing. This process is done in order to make the algorithm more efficient by discarding the words which does not affect the accuracy of the algorithm. This process is applied with the help of natural language processing commonly known as NLP. A library of python which deals with the textual analysis of the data. This process removes the duplicate words and make the length of the tweets shorter in order to minimize the training time and efficiency.

For applying this process, we have passed the tweets from the following process:

- 5.1. The text has been splitted into tokens i.e., into smaller fragments of words and stored in a data structure for further processing.
- 5.2. All the words from that data structure are then converted to the lower case for a better understanding by the algorithm.
- 5.3. Using the regular expression for the python library re, all the punctuations have been removed from the data structure containing words.
- 5.4. After the removal of the punctuations from the word's data structure, we have removed the non-alphabetic tokens like numbers or emojis as they will not contribute to the prediction and training.
- 5.5. Then further, we have only words in the data structure, so we have used stemmer in order to deviate the words to their base form. The data structure will be updated here.
- 5.6. At the end of the process, we have re-joined the words into sentences for the analysis purpose. The sentence will be formed in form of tweets as thy were original but in a clean way.

6. Model Training:

Model training is a process of making an algorithm learn what it must do further. A Machine learning model is train over a pre-defined dataset in which the output of the task was defined. The model training is performed by three types:-

- i. Supervised Learning
- ii. Unsupervised Learning
- iii. Reinforcement Learning

Training a model under supervision with the subject and its target in the dataset is known as supervised learning algorithm. In our model we have used five supervised learning algorithms to train and test the model. The model is trained through process which include the learning, error tuning, and then validation. The model first learns from the training dataset that which type of statement is negative, which are positive and rest neutral. It understands the negative and positive words and there affect with other words in a sentence.

The models which are used in the paper for the comparative analysis on accuracy and performance are as said below.

6.1. Logistic Regression: -

Logistic Regression is a classifier in the machine learning which is used to classify the data into distinct groups. This is a supervised machine learning algorithm. The paper Tyagi et al. , [31] describes how the logistic regression works with the equations used for the computation. The logistic regression predicts the dependent data using the dataset of independent variable which will be given to the algorithm. The logistic regression gives discrete value instead of binary output as 0 or 1. The discrete value given by the logistic regression is a probabilistic value of being something and lies between 0 and 1. The 0 and 1 are both the extreme values for the logistic regression so we use sigmoid function to map the value in this range. Sigmoid function is a function which is used by different machine learning algorithms. This function is basically used when we need to be given a value from negative infinity to positive infinity and the machine requires the value from a range of 0 to 1. This function maps the value by changing it to a range of 0 to 1 for further process.

$$\log \left[\frac{y}{y-1} \right] = b_0 + b_1 x_1 + b_2 x_2 + \dots + b_n x_n \quad (1)$$

The equation 1 will give the coordinates to be plotted on a plane on which the classification is done. To categorize the output in different category a threshold value is considered for the classification of the data which is in between the 0-1 range. Above the threshold value, the predicted value will be considered as one else zero.

6.2. Naïve Bayes: -

Naïve Bayes is a supervised learning algorithm which is used for the classification of the data. As mentioned in the paper Kewswun et al. , [32], Naïve Bayes uses the methodology of the popular bayes theorem which is used to find out the best hypothesis from a given space. Here the space is the dataset given to the algorithm. The bayes theorem states:

$$p(h/d) = \frac{p(d/h)*p(h)}{p(d)} \quad (2)$$

In the equation 2, ‘h’ is the best suited hypothesis on a given ‘d’ dataset. The ‘p(d/h)’ figures out the probability of ‘h’ with respect to ‘d’ . It uses probabilistic calculation for the prediction by quantifying the dataset into likelihood table of a hypothesis. It has three ways for the classification as listed below:

1. Gaussian Naïve Bayes
2. Multinomial Naïve Bayes
3. Bernoulli Naïve Bayes

The preferred model used is Gaussian Naïve Bayes algorithm which uses sci-kit-learn library of the python to implement it in the code. The Gaussian Naïve Bayes algorithm works on the stated mathematical equation 3:

$$p(x/y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right) \quad (3)$$

6.3. Support Vector Machine: -

Support Vector Machine commonly known as SVM is a supervised learning algorithm used for the classification of data. SVM is also used for the regression problem, but it is best suited for the classification problem. According to paper Furkan Rustam [33], SVM use extreme points i.e., known as vectors which create a hyperplane. These vectors are known as support vectors. Hyperplane is the best decision boundary on a n-dimensional plane. There are two types of support vector machine: -

1. Linear SVM
2. Non-linear SVM

The python library sklearn is used to implement the support vector machine. The sklearn is commonly known as the sci-kit-learn. In our project the paper uses multiple kernels to classify the data for a comparative analysis of different models. Kernel helps to gather the input data and transform the non-linear decision surface into a linear one in training dataset by using different mathematical equations. The kernel used for the model is listed below with the equations used:

6.3.1. Gaussian Kernel Radial Basis Function:

$$K(x, y) = e^{-\gamma \|x - y\|^2} \quad (4)$$

6.3.2. Sigmoid Kernel:

$$K(x, y) = \tanh(\gamma \cdot x^T y + r) \quad (5)$$

6.3.3. Polynomial Kernel:

$$K(x, y) = \tanh^d(\gamma \cdot x^T y + r)^d, \gamma > 0 \quad (6)$$

6.3.4. Linear Kernel:

$$f(x) = B(0) + \sum(ai * (x, y)) \quad (7)$$

6.4. Decision Tree Classifier: -

Decision Tree Algorithm is used to solve both category of the problem of machine learning both classification and regression. It is also a supervised machine learning algorithm. According to the paper Qi et al. , [34], this algorithm creates a decision tree based on the training data having three components: Root, Decision Node and Leaf Node as output. After the creation of the decision tree the model uses that tree to predict the data of the test set. Decision tree uses Attribute Selection Method for the choice of the best attribute from the data set to recursively generate a tree. This process repeats until it cannot make further tree. The ASM uses two famous techniques for choice of the attributes.

6.4.1. Information Gain:

It uses the quantity of change in the randomness in the data after the breaking of the data through segmentation based on the attribute. The mathematical expression used for the calculation of the Information Gain is.

$$I.G. = E(S) - [(W) * E(f)] \quad (8)$$

Here in the equation 8 E(S) is the Entropy of the sample, W is the Weighted average and E(f) is the Entropy of each feature. The E(S) is determined using a mathematical expression as shown below.

$$E(S) = -P(y)\log_2 P(y) - P(n)\log_2 P(n) \quad (9)$$

where P(y) is probability of being yes and as P(n) as probability of being no.

6.4.2. Gini Index:

It uses the quantity of being pure or impure during the creation of the tree. The lower the gini index the most preferable attribute it is. The gini index (G.I.) uses a mathematical expression for determining the G.I. as listed.

$$G.I. = 1 - \sum jPj^2 \quad (10)$$

The python library sklearn is used to implement the Decision Tree Algorithm. The sklearn is commonly known as the sci-kit-learn. Here we have set the criterion as entropy for best result.

6.5. Random Forest Algorithm: -

It is the most famous machine learning algorithm which comes under the category of the supervised learning algorithm used for both regression and classification model. It is mentioned in the paper Fakhrezi et al. , [35] that the basic concept of random forest algorithm is ensemble learning. The ensemble learning can be defined as the process of using multiple classifiers for a single complex classification problem. It combines different algorithm to improve the performance of the machine learning model. The name includes random forest which means that this algorithm is going to use a group of decision tree on the random basis. The prediction is done based on the average of the prediction of all the decision tree. The python library sklearn is used to implement the random forest algorithm. The sklearn is commonly known as the sci-kit-learn. The number of random trees is passed as an argument int the RandomForestClassifier class as a value of variable n_estimator. The value of the n_estimator is ten which is default, but we can override it by taking care of the overfitting of the data in that model.

6.6. KNN Algorithm: -

According to the paper Fatehi et al. , [36], the KNN stand for the Kth nearest neighbor which is categorized in the supervised learning algorithm and is used only for the classification problem. KNN at the training phase only store data and uses the data at the time of classification so it is called lazy learner algorithm. KNN does not make any assumption based on the underlying data hence it is called non-parametric algorithm. The dataset is stored and when the prediction dataset is run in the model, it iterates over each row of the data used in training set and check for the k nearest neighbor and according to that it predicts the output. To find the distance between two rows of the data, it uses Euclidean distance mathematical equation 11 as stated.

$$d = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \quad (11)$$

In the equation 11 d is the Euclidean distance X_1 and Y_1 are the training dataset attributes value and X_2 and Y_2 are the testing dataset attributes value. The KNN is also implemented by using the KNeighborsClassifier class from the sklearn library. The number of neighbors is passed as the argument of the class as the value of variable n_neighbour. We have used five neighbors for the classification of our model.

6.7. Gradient Boosting Machine: -

Gradient Boosting Machine also known as GBM uses a combination of multiple simple models to boost the performance of the model. According to the author of the paper Subramani et al. , [37], GBM takes the training dataset to make it fit in the model and then assign an equal weight to all the points of data. If the data point classified has an error, it reassigns the weight until it gets a suitable result. It is one of the best and powerful technique for both the classification and regression problem. GBM is also a supervised learning algorithm but depends on the average of the output of multiple simple algorithms. GBM consist of the listed element for the computation.

1. Loss Function
2. Weak Learner
3. Additive Model

It is an ensemble learning algorithm so implemented by the help of GradientBoostingClassifier from the sklearn. ensemble library.

7. Result and Discussion:

Model Prediction is a technique using Machine Learning and Data Mining to predict and forecast future results. It analyzes current and past data and scrutinize the learning model to forecast the favorable outcomes. We have split the dataset into two parts on a ratio of 8:2 with the use of train_test_split class from the sklearn library. The 80% of the dataset belongs to the training set and the rest 20% belongs to the testing set. We train the classifier with the training set and make prediction by using the testing set on the trained model using different classifiers. shows the comparison of the accuracies of different classification algorithms used in the research.

A comparative study of different algorithm is done in **Error! Reference source not found.**on dataset 1 which calculates the precision, recall and f1-score of all the category i.e., negative, positive, and neutral on each classifier used for the comparison. This helps in the analysis of the difference in the prediction of sentiments by using logistical regression, support vector machine with different kernels (i.e., sigmoid, polynomial, radial basis function and linear kernel), decision tree, random forest, knn and gradient boosting machine. In this the performance parameter of logistic regression compared to all other is better and has a precision of 0.89 in positive tweets, 0.74 in negative tweets and 0.85 in neutral tweets. While the performance parameter of Gradient Boost Machine is the worst among all the algorithms having a precision of 0.87 in positive, 0.67 in negative and 0.51 in neutral. In positive tweets the precision of the Support Vector Machine with RBF as kernel is highest as 0.94 while for the negative and neutral tweets the precision of Logistic Regression as 0.74 and KNN as 0.89 respectively.

Table 2 Performance parameter of each algorithm on twitter dataset 1.

Algorithms		Positive			Negative			Neutral		
		Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Support Vector Machine	Linear	0.88	0.83	0.85	0.72	0.84	0.77	0.84	0.79	0.82
	Polynomial	0.38	0.97	0.55	0.44	0.03	0.06	0.87	0.38	0.53
	Sigmoid	0.92	0.76	0.83	0.65	0.82	0.73	0.82	0.79	0.80
	RBF	0.94	0.74	0.83	0.72	0.80	0.76	0.81	0.88	0.84
Decision Tree		0.84	0.77	0.80	0.68	0.76	0.72	0.77	0.76	0.77
Random Forest		0.87	0.75	0.81	0.61	0.82	0.70	0.81	0.70	0.75
KNN		0.69	0.80	0.74	0.46	0.92	0.61	0.89	0.19	0.31
Gradient Boost Machine		0.87	0.86	0.91	0.67	0.87	0.89	0.41	0.58	0.54
Logistic Regression		0.89	0.83	0.86	0.74	0.84	0.79	0.85	0.82	0.84

After the study on dataset one, comparison is done in Table 3 by the performance parameters of each algorithm using twitter dataset 2. The performance parameter of logistic regression compared to all other is better and has a precision of 0.69 in positive tweets, 0.79 in negative tweets and 0.77 in neutral tweets. While the performance parameter of Gradient Boost Machine is the worst among all the algorithms having a precision of 0.78 in positive, 0.32 in negative and 0.95 in neutral. Although the performance of Gradient boost machine in the neutral tweets is best but due to the precision score of its negative tweets the average of precision of each of the sentiments is 0.68 which make it worst.

Table 3 Performance parameter of each algorithm on twitter dataset 2

Algorithms		Positive			Negative			Neutral		
		Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Support Vector Machine	Linear	0.67	0.74	0.70	0.77	0.78	0.77	0.74	0.68	0.71
	Polynomial	0.35	0.95	0.51	0.83	0.41	0.54	0.91	0.17	0.28
	Sigmoid	0.53	0.69	0.60	0.61	0.75	0.67	0.71	0.44	0.54
	RBF	0.57	0.79	0.66	0.69	0.80	0.74	0.74	0.57	0.65
Decision Tree		0.60	0.61	0.60	0.65	0.63	0.64	0.58	0.63	0.60
Random Forest		0.61	0.62	0.62	0.62	0.70	0.66	0.62	0.59	0.60
KNN		0.49	0.50	0.49	0.57	0.57	0.57	0.45	0.49	0.47
Gradient Boost Machine		0.78	0.13	0.22	0.32	0.99	0.49	0.95	0.04	0.08
Logistic Regression		0.69	0.79	0.74	0.79	0.82	0.80	0.77	0.71	0.74

At last, the work uses a vast number of tweets i.e., 1600000 for the analysis of the performance of the six algorithms. The result of the analysis of the dataset 3 is shown in Table 4 as precision, recall and f1-score. For the positive tweets among the dataset Logistical Regression achieved the highest precision of 0.94. For the negative tweets, the highest precision is achieved by Logistic Regression again as 0.89. Support Vector Machine with polynomial kernel give the highest precision for the neutral tweets as 0.94. For overall precision logistic regression is best among all and has a precision of 0.94 in positive tweets, 0.89 in negative tweets and 0.86 in neutral tweets. On the other hand, Support Vector Machine with polynomial kernel achieved the worst precision of an average 0.71.

Table 4 Performance parameters of each algorithm on dataset3.

Algorithms		Positive			Negative			Neutral		
		Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Support Vector Machine	Linear	0.92	0.91	0.87	0.87	0.84	0.89	0.91	0.87	0.89
	Polynomial	0.75	0.88	0.94	0.48	0.84	0.89	0.91	0.88	0.94
	Sigmoid	0.91	0.84	0.76	0.85	0.88	0.87	0.86	0.81	0.84
	RBF	0.93	0.87	0.87	0.88	0.84	0.91	0.88	0.87	0.83
Decision Tree		0.88	0.99	0.93	0.87	0.81	0.84	0.90	0.83	0.86
Random Forest		0.90	0.86	0.88	0.83	0.69	0.75	0.83	0.96	0.89
KNN		0.92	0.25	0.40	0.74	0.18	0.28	0.42	0.99	0.59
Gradient Boost Machine		0.92	0.86	0.75	0.68	0.95	0.75	0.91	0.89	0.89
Logistic Regression		0.94	0.90	0.92	0.89	0.75	0.82	0.86	0.98	0.92

After the prediction made by the different machine learning model, we can analyze the performance of each of the classifiers used in the machine learning model. **Error! Reference source not found.** gives a brief idea of the performance of the classifiers. In the table 4, a comparative study of 6 supervised machine learning on three different datasets are illustrated. For dataset 1, the accuracy of logistic regression is 83.15 which is best among all six algorithms while the accuracy of gradient boost machine is 41.09 which is worst. For dataset 2, the accuracy of logistic regression is 74.97 which is again best among all six algorithms while the accuracy of gradient boost machine is 35.10 which is again the least. Now for dataset 3, the logistic regression again performs the best with highest accuracy of 89.94 among all datasets and algorithms while KNN performs the worst having an accuracy of 49.73. On an average of all the datasets, the Logistical regression remains at the position of the best performer having an average accuracy of 82.68 and the gradient boost machine performs worst having an average accuracy of 51.64.

Table 5 Accuracy of each algorithm on different datasets

No.	Algorithms	Dataset 1	Dataset 2	Dataset 3	Average
1	Logistic Regression	83.15	74.97	89.94	82.68
2	Support Vector Machine	Linear	78.65	81.46	71.35
3		Polynomial	83.84	47.15	45.55
4		Sigmoid	81.59	78.87	58.84
5		RBF	81.86	81.46	72.87
6		Decision Tree	76.20	66.07	76.46
7	Random Forest	75.07	61.12	88.52	74.69
8	KNN	58.15	49.62	49.73	52.50
9	Gradient Boost Machine	41.09	35.10	78.68	51.64

For a brief analysis of the prediction, confusion matrix is used. The confusion matrix used in this project is obtained with the help of python library. From the analysis of dataset 3 which contains 1600000 tweets, Figure 2 shows the confusion matrix of Logistical Regression which gives a brief analysis of the true prediction labeled on the y-axis and the predicted value labeled on the x-axis. The confusion matrix is obtained after the model predicts the output and the prediction is compared to the original results. The confusion matrix is used to get the performance of the algorithm. In confusion matrix, the number of tweets whose sentiments are predicted correctly are placed in the diagonal of the matrix. The 0, 1 and 2



Fig. 2. Confusion Matrix of Logistical Regression for dataset3

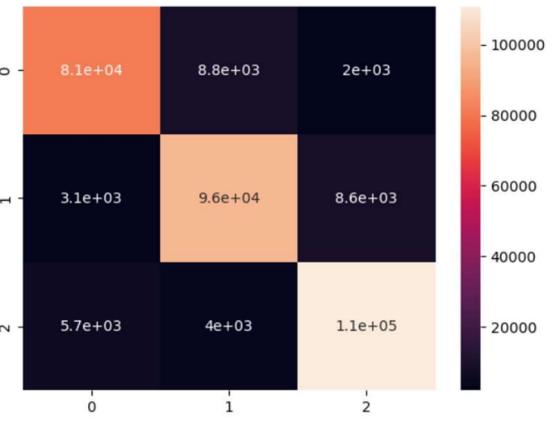


Fig. 3. Confusion Matrix of Decision Tree for dataset3

in the axis denotes the Negative, Neutral and Positive tweets classification.

To compare between different algorithms, we have three parameters here precision, recall and f1-score. The mathematical formula for calculating each are as follows:

$$precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (12)$$

$$recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (13)$$

$$f1 - score = 2 * \frac{precision * recall}{precision + recall} \quad (14)$$

So, from the confusion matrix we can analyze that the sum of correct predictions is 287808 (72214 negative, 90262 neutral and 120788 positive) out of 320000 tweets. This gives an accuracy of 89.94% which is highest, among others. The other parameters are also derived from the confusion matrix.

Similarly, the Figure 3 also shows the confusion matrix of Decision Tree providing a brief idea of the overall prediction of sentiments. In this confusion matrix, the correctly predicted tweets are 283264 out of 320000 tweets with an accuracy of 88.52%. After the overall analysis of the different model, we have analyzed that the best model which fits to our dataset for the classification is Logistic Regression. The runner up for the dataset is the Decision Tree Algorithm. This can change for the different dataset or

different problem because the machine learning is all about the prediction of the data.

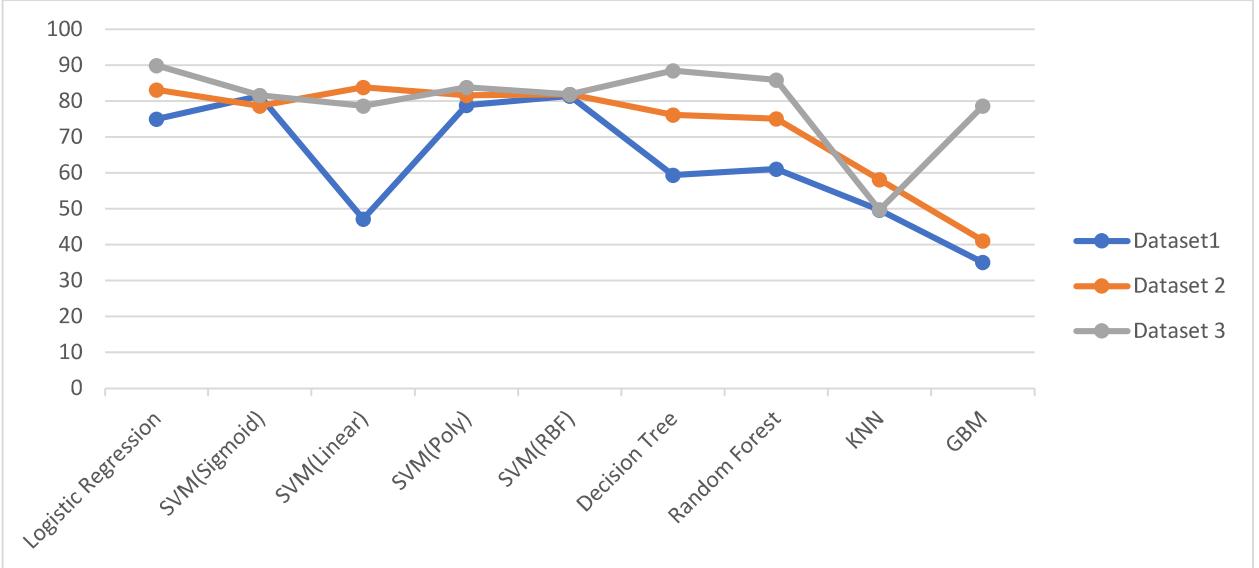


Fig. 4. Line Graph of the accuracy of each algorithm on each of the datasets used in this project.

Figure 4 Line Graph of the accuracy of each algorithm used in this project shows a plot of the accuracy of different algorithms on a linear graph for a visualized idea of the performance of each classifier.

8. Conclusion:

After the prediction of sentiments of each of the three datasets by using 6 different supervised machine learning algorithms, the paper concludes that the highest accuracy for the prediction of sentiment is achieved by Logistic Regression. The accuracy of Logistic regression is 83.15 for dataset 1, 74.97 for dataset 2 and 89.94 for dataset 3. The lowest accuracy is achieved by the gradient boost machine for dataset 1 and dataset 2 as 41.09 and 35.10 respectively. As we increase the size of the dataset the accuracy of gradient boost machine increases to 78.68, which demonstrate that the algorithm gradient boost machine works better for a large dataset (here dataset 3 consisting of 1600000 tweets). For dataset 3, KNN performs worst showing an accuracy of 49.73. The findings of the work demonstrated that with an increase in the size of dataset used for the training and prediction, the number of features increases which led to an increase in the efficiency.

9. Future Scope:

Tweeter is a famous platform for opinion sharing about any topic, product or anything going on in mass. So, to get an idea about anything going in the public people uses twitter and read the opinion about that topic. In recent years, most of the public uses twitter for sharing their review and opinion over any protest going on, product launched by any company or any public problem so that their opinion can reach in mass. The analysis we have done here classifies some of the best algorithms for sentiment analysis of any textual opinion.

The model trained here can be used for the analysis of the sentiment of the public or mass about any product in commercial field so that the company can extract the negative opinion among all the opinions and enhance the product. This will give a boost in the marketing of the product and helps the company to find out the negativity in the product. In commercial area, when a company wants to introduce any new product, he can analyze the feedback of the similar past product to take a brief idea of the opinion of the public.

Similarly, in political context by designing a UI a person can get the opinion about anyone or any party to analyze their image in public. One drawback of our algorithm is that we have used classifiers for the analysis of sentiments. The improvement can be done by using reinforcement learning algorithms by implementing TensorFlow library from python. TensorFlow uses perceptron model to predict the result giving a better accuracy with a less training time needed.

References:

- [1] Z. Drus and H. Khalid, "Sentiment analysis in social media and its application: Systematic literature review," in *Procedia Computer Science*, 2019, vol. 161, pp. 707–714. doi: 10.1016/j.procs.2019.11.174.
- [2] V. A. Kharde and S. S. Sonawane, "Sentiment Analysis of Twitter Data: A Survey of Techniques," 2016. [Online]. Available: <http://ai.stanford.edu>.
- [3] A. Agarwal, O. Rambow, R. J. Passonneau, B. Xie, I. Vovsha, and R. Passonneau, "Sentiment Analysis of Twitter Data ConEdison View project Morphologically Annotated Corpora and Morphological Analyzers for Moroccan and Sanaani Yemeni Arabic View project Sentiment Analysis of Twitter Data," 2011. [Online]. Available: <https://www.researchgate.net/publication/247935218>
- [4] P. Sudhir and V. D. Suresh, "Comparative study of various approaches, applications and classifiers for sentiment analysis," *Global Transitions Proceedings*, vol. 2, no. 2, pp. 205–211, Nov. 2021, doi: 10.1016/j.gltip.2021.08.004.
- [5] K. Dhola and M. Saradva, "A comparative evaluation of traditional machine learning and deep learning classification techniques for sentiment analysis," in *Proceedings of the Confluence 2021: 11th International Conference on Cloud Computing, Data Science and Engineering*, Jan. 2021, pp. 932–936. doi: 10.1109/Confluence51648.2021.9377070.
- [6] A. S. Neogi, K. A. Garg, R. K. Mishra, and Y. K. Dwivedi, "Sentiment analysis and classification of Indian farmers' protest using twitter data," *International Journal of Information Management Data Insights*, vol. 1, no. 2, Nov. 2021, doi: 10.1016/j.ijimei.2021.100019.
- [7] A. B. S, R. D. B, and R. M. Kumar, "Real Time Twitter Sentiment Analysis using Natural Language Processing." [Online]. Available: www.ijert.org
- [8] S. Singh, "Twitter Sentiments Analysis Using Machine Learning," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, pp. 312–320, Jul. 2020, doi: 10.32628/cseit2020456.

- [9] Faizan, "Twitter Sentiment Analysis," 2019. [Online]. Available: www.ijisrt.com
- [10] V. Jain and M. Parmar, "A Review on Emotion and Sentiment analysis Using Learning Techniques," *Int J Res Appl Sci Eng Technol*, vol. 10, no. 11, pp. 395–405, Nov. 2022, doi: 10.22214/ijraset.2022.47337.
- [11] P. Nandwani and R. Verma, "A review on sentiment analysis and emotion detection from text," *Social Network Analysis and Mining*, vol. 11, no. 1. Springer, Dec. 01, 2021. doi: 10.1007/s13278-021-00776-6.
- [12] A. Mitra, "Sentiment Analysis Using Machine Learning Approaches (Lexicon based on movie review dataset)," *Journal of Ubiquitous Computing and Communication Technologies*, vol. 2, no. 3, pp. 145–152, Sep. 2020, doi: 10.36548/jucct.2020.3.004.
- [13] R. Liu, Y. Shi, C. Ji, and M. Jia, "A Survey of Sentiment Analysis Based on Transfer Learning," *IEEE Access*, vol. 7. Institute of Electrical and Electronics Engineers Inc., pp. 85401–85412, 2019. doi: 10.1109/ACCESS.2019.2925059.
- [14] L. Nemes and A. Kiss, "Social media sentiment analysis based on COVID-19," *Journal of Information and Telecommunication*, vol. 5, no. 1, pp. 1–15, 2021, doi: 10.1080/24751839.2020.1790793.
- [15] H. T. Phan, V. C. Tran, N. T. Nguyen, and D. Hwang, "Improving the Performance of Sentiment Analysis of Tweets Containing Fuzzy Sentiment Using the Feature Ensemble Model," *IEEE Access*, vol. 8, pp. 14630–14641, 2020, doi: 10.1109/ACCESS.2019.2963702.
- [16] M. Hoang, O. Alija Bihorac, and J. Rouces, "Aspect-Based Sentiment Analysis Using BERT."
- [17] A. Yadav and D. K. Vishwakarma, "Sentiment analysis using deep learning architectures: a review," *Artif Intell Rev*, vol. 53, no. 6, pp. 4335–4385, Aug. 2020, doi: 10.1007/s10462-019-09794-5.
- [18] A. Alsaedi and M. Z. Khan, "A study on sentiment analysis techniques of Twitter data," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 2, pp. 361–374, 2019, doi: 10.14569/ijacsa.2019.0100248.
- [19] A. M. John-Otumu, M. M. Rahman, O. C. Nwokonkwo, and M. C. Onuoha, "AI-Based Techniques for Online Social Media Network Sentiment Analysis: A Methodical Review," *International Journal of Science, Engineering and Technology*, vol. 16, no. 12, 2022.
- [20] S. Aslan, S. Kızıloluk, and E. Sert, "TSA-CNN-AOA: Twitter sentiment analysis using CNN optimized via arithmetic optimization algorithm," *Neural Comput Appl*, 2023, doi: 10.1007/s00521-023-08236-2.
- [21] H. Tian *et al.*, "SKEP: Sentiment Knowledge Enhanced Pre-training for Sentiment Analysis," May 2020, [Online]. Available: <http://arxiv.org/abs/2005.05635>

- [22] S. Zad, M. Heidari, J. H. Jones, and O. Uzuner, "A Survey on Concept-Level Sentiment Analysis Techniques of Textual Data," in *2021 IEEE World AI IoT Congress, AlloT 2021*, May 2021, pp. 285–291. doi: 10.1109/AlloT52608.2021.9454169.
- [23] K. H. Manguri, R. N. Ramadhan, and P. R. Mohammed Amin, "Twitter Sentiment Analysis on Worldwide COVID-19 Outbreaks," *Kurdistan Journal of Applied Research*, pp. 54–65, May 2020, doi: 10.24017/covid.8.
- [24] A. D. Dubey Assistant Professor, "Twitter Sentiment Analysis during COVID19 Outbreak." [Online]. Available: <https://ssrn.com/abstract=3572023>
- [25] K. R. Mabokela, T. Celik, and M. Raborife, "Multilingual Sentiment Analysis for Under-Resourced Languages: A Systematic Review of the Landscape," *IEEE Access*, pp. 1–1, 2022, doi: 10.1109/ACCESS.2022.3224136.
- [26] L. Yang, Y. Li, J. Wang, and R. S. Sherratt, "Sentiment Analysis for E-Commerce Product Reviews in Chinese Based on Sentiment Lexicon and Deep Learning," *IEEE Access*, vol. 8, pp. 23522–23530, 2020, doi: 10.1109/ACCESS.2020.2969854.
- [27] V. A. Kharde and S. S. Sonawane, "Sentiment Analysis of Twitter Data: A Survey of Techniques," 2016. [Online]. Available: <http://ai.stanford.edu/>
- [28] "Search | Kaggle." <https://www.kaggle.com/datasets/yasserh/twitter-tweets-sentiment-dataset?resource=download> (accessed Mar. 02, 2023).
- [29] "Twitter Sentiment Analysis | Kaggle." <https://www.kaggle.com/datasets/jp797498e/twitter-entity-sentiment-analysis> (accessed Mar. 02, 2023).
- [30] "1.6 Million Dataset of Twitter."
- [31] A. Tyagi and N. Sharma, "Sentiment Analysis using logistic regression and effective word score heuristic," *International Journal of Engineering and Technology(UAE)*, vol. 7, no. 2, pp. 20–23, 2018, doi: 10.14419/ijet.v7i2.24.11991.
- [32] N. Kewsuwan and S. Kajornkasirat, "A sentiment analysis model of agritech startup on Facebook comments using naive Bayes classifier," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 3, pp. 2829–2838, Jun. 2022, doi: 10.11591/ijece.v12i3.pp2829-2838.
- [33] Furqan Rustam, Madiha Khalid, Waqar Aslam, Vaibhav Rupapara, Arif Mehmood, and Gyu Sang Chol, "A performance comparison of supervised machine learning models for Covid-19 tweets sentiment analysis", Accessed: Mar. 02, 2023. [Online]. Available: <http://doi.org/10.1371/journal.pone.0245909>
- [34] Y. Qi and Z. Shabrina, "Sentiment analysis using Twitter data: a comparative application of lexicon- and machine-learning-based approach," *Soc Netw Anal Min*, vol. 13, no. 1, Dec. 2023, doi: 10.1007/s13278-023-01030-x.

- [35] M. F. Fakhrezi, Adian Fatchur Rochim, and Dinar Mutiara Kusomo Nugraheni, "Comparison of Sentiment Analysis Methods Based on Accuracy Value Case Study: Twitter Mentions of Academic Article," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 1, pp. 161–167, Feb. 2023, doi: 10.29207/resti.v7i1.4767.
- [36] N. Fatehi, M. A. Motlagh, and · Hadishahriar Shahhoseini, "A Reliable Sentiment Analysis for Classification of Tweets in Social Networks." [Online]. Available: <https://www.researchgate.net/publication/365806253>
- [37] N. Subramani, S. Veerappampalayam Easwaramoorthy, P. Mohan, M. Subramanian, and V. Sambath, "A Gradient Boosted Decision Tree-Based Influencer Prediction in Social Network Analysis," *Big Data and Cognitive Computing*, vol. 7, no. 1, p. 6, Jan. 2023, doi: 10.3390/bdcc7010006.