SENTIMENT ANALYSIS AND OFFENSIVE DETECTION FROM TWEETS WITH MACHINE LEARNING APPROACH

BY

Syeda Suria Fatema
ID: 0242310005103040

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Masters of Science in Computer Science and Engineering.

Supervised By

Abdus Sattar

Assistant Professor & Coordinator MSc
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

JULY 06,2024

APPROVAL

This Thesis titled "Sentiment Analysis and Offensive Detection from Tweets with Machine Learning Approach", submitted by Syeda Suria Fatema, ID No:0242310005103040 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of M.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 06-07-2024.

BOARD OF EXAMINERS

Chairman

Narayan Ranjan Chakraborty
Associate Professor and Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Md. Sadekur Rahman
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Dr. Ohidujjaman, PhD
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

External Examiner

Mr. Sadat Hossain
Data Scientist
Risk Management Division,
BRAC Bank Limited

DECLARATION

I hereby declare that, this project has been done by us under the supervision of **Abdus Sattar** , Assistant Professor & Coordinator MSc, **Department of CSE** Daffodil International University. We also declare that neither this thesis nor any part of this thesis has been submitted elsewhere for award of any degree or diploma.

Supervised by:

Abdus Sattar
Assistant Professor & Coordinator MSc
Department of CSE
Daffodil International University

Submitted by:

Syeda Suria Fatema
ID:0242310005103040
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

I am begin by extending my heartfelt gratitude to the Almighty for blessing me and enabling me to successfully complete my final year thesis.

My sincere appreciation goes to **Abdus Sattar ,** Assistant Professor & Coordinator MSc of the Department of Computer Science and Engineering at Daffodil International University, Dhaka. Their extensive knowledge and keen interest in the field of "Natural Language Processing" were instrumental in guiding me throughout this thesis. Their unwavering patience, scholarly guidance, continuous encouragement, dedicated supervision, constructive criticism, valuable advice, and thorough review of multiple drafts at every stage played a crucial role in the completion of this thesis.

I also like to express our deepest thanks to **Dr. Sheak Rashed Haider Noori**, the Professor & Head of the Department of CSE, for his invaluable assistance in bringing my thesis to fruition. My gratitude extends to all the faculty members and staff of the CSE department at Daffodil International University.

I am grateful to my fellow classmates at Daffodil International University, who engaged in discussions and provided support during the course of my thesis.

Lastly, we would like to acknowledge and show our utmost respect for the unwavering support and patience of our parents.

ABSTRACT

The emergence of objectionable content in the internet era has become a major worry, especially for groups speaking English. The focus of this work is on using machine learning methods to identify objectionable words in English. A machine learning model is trained on a particular English dataset that contains examples of both offensive and non-offensive material. The dataset is preprocessed, tokenized, and evaluated to create sequences that can be fed into a cutting-edge machine learning method. The dataset is used to train the machine learning-based model, which then uses techniques like word tokenization and TF-IDF to improve performance. To evaluate the efficacy of the model, assessment criteria like as accuracy, precision, recall, and F-1 score are utilized. Several strategies, including as data augmentation techniques and language-specific preprocessing techniques, are investigated to solve issues unique to English language offensive detection. The results demonstrate how well machine learning works to identify objectionable content in English, indicating the technology's potential to address this issue in settings where English is the primary language. The analysis demonstrates the Random Forest's better performance, showing that it obtained an accuracy of 96.04%. A other strategy that made use of Multi-Layer Perceptrons (MLP) also produced an accuracy of 95.865%. Additionally, ensemble models such as Bagging, Boosting, and Voting were used; their performance was optimized by hyper-parameter tweaking. Notably, recent experimental investigations of these data indicated that the Random Forest and MLP performed better than other models, achieving the greatest accuracy rate of 96.04% and 95.865% in offensive detection.

Keywords: Random Forest, MLP, Machine Learning, Algorithms, Ensemble Model, TF-IDF, Tokenizer.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Offensive content has become more commonplace in the digital age, impacting people of all ages and social classes. Social media and other online platforms have created new channels for destructive conduct, which has a negative effect on victims' mental, emotional, and social wellbeing. The English-speaking population has particular offensive issues because of language quirks and cultural influences that determine the character and expression of harmful online activities. Encouraging a safe and welcoming digital space for people using English to communicate requires identifying and addressing objectionable content. To develop a machine learning-based approach for recognizing offensive terms in English is the aim of this project. Machine learning has shown promise in a number of natural language processing applications and has the ability to accurately detect and categorize inappropriate content. Our objective is to further the development of offensive detection methods that are especially designed for the English language by utilizing machine learning models. Developing and refining a machine learning model that can consistently distinguish between English texts that are offensive and those that are not is the primary objective. In order to do this, we utilize an English dataset that contains both offensive and non-offensive text examples. The dataset is carefully preprocessed, tokenized, and vectorized into sequences that may be fed into a cutting-edge machine learning technique. In addition to training the model, we investigate other approaches to tackle problems in English offensive detection, such as language-specific preprocessing techniques, data augmentation strategies, and the inclusion of English-specific contextual and cultural elements. The purpose of this study's findings is to provide light on how well machine learning techniques work for identifying derogatory words in English. Additionally, this study aids in the creation of preventative strategies to counteract offensive and foster a safer online environment. Overall, the study clarifies the unique difficulties and possibilities related to identifying offensive language in the English

language, acting as a first step in resolving this important problem among communities of English speakers.

1.2 Motivation

The motivation behind doing this study on machine learning-based offensive detection for the English language is based on many key factors. The most important of them is the realization that offensive behavior has grown to be a significant social issue that affects people of all ages, from kids to adults. The urgent need to address offensive behavior is highlighted by the well-established negative consequences it has on mental health, self-esteem, and general well-being. Unfortunately, most of the research that has been done on offensive detection has focused on the English language, which has left a gap in our knowledge of how to address this issue in English-speaking environments. Like many other English-speaking groups, the English-speaking community has unique difficulties when it comes to objectionable. Because of the peculiarities of the English language, cultural influences, and common online behaviors among English-speaking people, customized strategies are required to recognize and address objectionable content in this particular environment. This study creates a machine learning-based model for English language offensive detection in an effort to bridge the research gap and offer valuable insights on handling offensive in English-speaking contexts. The expected results of this research might make it easier to develop language-specific tools, tactics, and interventions that are intended to detect, stop, and lessen offensive episodes in the online English community. Furthermore, this study aims to promote inclusion, guarantee that people speaking English receive fair attention, and shield them from abusive words by focusing on the English language. This study lays the foundation for future research and activities that address the needs of varied language groups by highlighting the importance of linguistic variety and cultural sensitivity in the fight against online abuse. The goal of this project is to provide a more secure and welcoming online environment for those who use English as their first language. By employing machine learning methodologies and addressing language-specific challenges, this study seeks to make a substantial contribution to the overarching goal of deterring offenses and advancing a dignified and welfare-focused digital society.

1.3 Rationale of the Study

There are several reasons for this research, but the main one is to fill in important gaps in the literature on offensive detection in English-speaking situations. The goal of the study is to advance our knowledge of offensiveness by exploring the linguistic and cultural nuances present in the English language. By doing this, it hopes to strengthen English-speaking communities and promote the creation of solutions specifically tailored to this language group. This study's investigation of machine learning in relation to natural language processing is a key component. The research attempts to ascertain the effectiveness of these advanced methods inside the complex structure of the English language by use of machine learning techniques. This investigation broadens the scope of machine learning applications and provides new perspectives on how well it works to handle language-specific issues related to offense. Moreover, a dedication to promoting a safer digital world is central to our study. Through its emphasis on English-speaking people, the research promotes the development of safeguards that account for linguistic diversity. It is essential to place this focus on linguistic and cultural subtleties in order to create a digital environment that is safe and welcoming to everyone, regardless of the language they speak. Essentially, the logic for the research is based on bridging gaps, comprehending cultural subtleties, empowering communities, investigating cutting-edge technology, and eventually helping to create a safer online environment for people who are using the English language for communication.

1.4 Research Question

1. What is the likelihood that a statement will be classified as either offensive or not?

2. How difficult is it to tell if anything someone has said something offensive?

3. What advantages does our suggested model provide?

4. In what ways can we evaluate how well our model detects objectionable content?

5. How much effectiveness do the algorithms used in our suggested model exhibit?

1.5 Expected output

With an emphasis on the English language, this study aims to close the current gap in offensive detection studies for English situations. Through exploring the domain of offense in the English-speaking community, the research seeks to provide customized solutions that successfully tackle the particular difficulties encountered by this language community. To effectively prevent online harassment, certain techniques and interventions are required due to the unique cultural and linguistic peculiarities inherent for English-Speaking community. We focused to create a model that can correctly identify objectionable material in English texts by using machine learning techniques, specifically a machine learning model. The study's expected results will aid in the creation of language-specific instruments and guidelines intended to stop and lessen offensive situations. The ultimate goal is to foster an online community that is safer and more welcoming for those who use the English language for communication.

1.6 Project Management and Finance

In our technique offers an affordable solution that can be applied on a daily basis and has a great deal of potential as a useful tool for offensive detection in our nation. While high-configuration tools can improve performance and produce remarkable results, the actual implementation of this detection procedure necessitates the use of readily available tools. However, our strategy continues to be successful even with simpler tools, guaranteeing smooth operation and making a substantial contribution to the continuing operations against aggressive.

1.7 Report Layout

The proposed study is partitioned into many important segments like: first one covers the background research in detail, going into the offensive setting and pertinent literature. A thorough description of the methods, instruments, and strategies used to carry out the investigation and create the suggested model is given in the research methodology section. The study results, conclusions, and a thorough explanation of the findings are then presented in the experimental results and discussion section. A summary that draws

conclusions and provides ideas for further research completes the study. The literature and sources that were used to support the study's conclusions are included at the section of references.

CHAPTER 2

BACKGROUND STUDY

2.1 Preliminaries

Analyzing attack patterns precisely requires the application of machine learning techniques. In this section, we will examine studies concerning the assessment of social media commentary reports. For this study, a variety of computational models are used, including Bagging, Adaboost, Gaussian Naïve Bayes, MLP, Random Forest, Decision Tree, XGBoost, KN Neighbors, Logistic Regression, and Voting. A major focus of the research activities in this department is machine learning models. The section also highlights a number of scholars that have used a range of models in their own investigations, which has improved our knowledge of offensive dynamics.

2.2 Related Works

Our machine learning algorithms are quite successful in the field of English language offensive detection. A thorough examination of research papers and techniques demonstrates a wide range of strategies used to fully handle the complex problems related to offensive. These approaches make use of a range of machine learning algorithms and natural language processing (NLP) techniques, each of which brings unique benefits and insights to the overall objective. Firstly, machine learning models become extremely useful tools when trying to find offending instances. They do this by using algorithms based on decision trees, or "tree structures," which facilitate efficient decision-making processes that are essential for precise detection [2].

The current uses of sentiment analysis across several fields are covered in this section. Using Twitter data, Malik et al. [3] used sentiment analysis in the viewing domain to forecast viewers of streaming TV episodes. The writers gathered tweets utilizing a certain keyword for a predetermined amount of time for every TV show.Additionally, they investigated how well different supervised machine learning techniques performed in classifying the emotions seen in tweets regarding TV shows. El Rahman et al. [4]

conducted a sentiment analysis on Twitter data related to the food retail sector in order to ascertain the popularity of KFC against McDonald's [5]. They used a variety of techniques for supervised categorization. Among the companies under investigation, Maxtent performed very well for KFC and McDonald's, obtaining a four-fold cross-validation of 78% for KFC and 74% for McDonald's [6].

Neogi et al. [7] conducted a sentiment analysis on the demonstrations by Indian farmers, based on the views expressed by the general public on Twitter, in the context of food supply. Their study assessed public sentiment in India over farmer protests using a lexicon-based approach called TextBlob in conjunction with an investigation of machine learning classification techniques. TF-IDF was employed in their modeling simulation as a feature extractor. Among the classifiers they looked at for their study, Random Forest was determined to have the best accuracy. A sentiment analysis of Twitter data related to certain retail phone brands was carried out by Hasan et al. [8]. To find out which phone brand—the Samsung or the iPhone—is more popular, the writers collected a thousand tweets.

To generate features for modeling, the authors looked at the use of term frequency-inverse document frequency (TF-IDF) and bag-of-words (BoW) [9]. It was evident from their research that the iPhone was more popular than Samsung devices as both TF-IDF and BoW boosted the performance of the results.Health-wise, Vijay et al. [10] looked at sentiment analysis of Twitter data related to COVID-19 in a few areas of India between November 2019 and May 2020 to evaluate how the virus spread, how people felt about it, and how the government handled it. During that period, TextBlob was used to examine Twitter data that was gathered for their study.

In the political domain, Elbagir et al. [11] classified tweets from the 2016 US elections into four categories: positive, negative, neutral, highly negative, and highly positive. They did this by using Natural Language Toolkit (NLTK) and Valence Aware Dictionary and Sentiment Reasoner (VADER) [12]. But Joyce et al. [13] also performed sentiment analysis during the US elections of 2016 in order to contrast two sentiment analysis techniques: lexicon-based sentiment analysis and machine learning classifiers for

sentiment. Their approach used both automatic and human tagging of tweets. Their results show that the lexicon-based sentiment analysis performed better [14].

The author [15] looked at the 2020 US presidential election to forecast the result and collect sentiment analysis between the two contenders on Twitter. In order to assess performance, five machine learning categories were examined in the data. Of the five classifiers, the multilayer perceptron (MLP) had the best performance. The sentiment analysis method was developed by Kaur et al. [16] to get opinions from people on Twitter on changes in power in both developed and developing nations. In their study, they proposed the use of lexicon-based VADER technique together with classification algorithms to infer sentiment polarity. TF-IDF was one method of feature extraction. We looked into four algorithms in order to assess performance. Random forests and decision trees were the two most successful models. [17].

2.3 Comparative Analysis and Summary

It required a significant amount of work to navigate the world of machine learning models, particularly in light of the increased demand for these models. Several concurrent studies faced difficulties, seeing suboptimal model performances and limited precision. Adopting a distinct machine learning model was necessary to reach the highest level of dataset detection accuracy. The project required the installation of state-of-the-art hardware infrastructure in order to guarantee the effective operation of these models. The approach involved performing laborious calculations to evaluate classification rates, and the use of expensive GPUs to facilitate the execution of complex models, but potentially resulting in a longer runtime.

Table 2.1 Comparative Analysis

Author	Used Algorithms	Best Results
Maxtent et al.	Random Baseline	78%
Neogi et al.	NB, DT, RF, SVC	RF=95.62%
Hasan et al.	TF-IDF	85.25%

Elbagir et al.	VADER	46%
Joyce et al.	Smoothing	94%
Kaur et al.	NB, DT, RF, LR	RF=84%
Our Algorithms	RF, DT, XGBoost, LR	RF=96.04%

2.4 Scope of the Problem

The problem of offensive language in the English language is complicated because it involves linguistic barriers, cultural influences, different internet platforms, and a heterogeneous population. To fully understand and address these aspects, a thorough strategy is required. Holistic solutions may be developed by considering the nuances of language, cultural dynamics, and the wide range of internet platforms. This, in turn, makes it easier to lessen inappropriate incidences and promotes a safer online space for those who communicate in English.

2.5 Challenges

The task of identifying offensive content in English is complicated by language complexities, cultural subtleties, changing online environments, and the ever-changing nature of offensive behavior. In order to overcome these challenges, algorithms must be created that are language-specific, take into account cultural settings, adjust to platform changes, and stay up to date with new offending tendencies. In order to successfully identify and stop offensive language usage, it is imperative that these issues are resolved. This will help create a safer online space for those who use English for communication.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Research Subject and Instrumentation

In order to get the best accuracy possible with our dataset, we used a variety of hybrid models and algorithms. Cutting-edge hardware tools, especially high-end GPUs, and the use of the Python programming language and its resources were essential to our success. Tools like Jupyter Notebook, Google Colab, and Anaconda were essential to our workflow since they were essential to our data processing and model training procedures. Python's vast library ecosystem and adaptability made code creation and execution easy, resulting in productive operations. Furthermore, browser-based coding features improved accessibility and collaborative activities. Examples of these platforms include Jupyter Notebook, Google Colab, and Anaconda. By utilizing this all-inclusive combination of tools and resources, we were able to push the limits of dataset correctness and provide reliable findings in our quest for perfection.

3.2 Data Collection Procedure

After acquiring the dataset from Kaggle [1], it was nearly ready for use, with the exception of two essential parts: the textual data and the labels that went with it. Three separate categories were included in this supervised dataset, which gave rise to a thorough representation of the content. The dataset was split into a test set and a training set in order to prepare it for model assessment. This division designated 80% of the data for in-depth analysis and training, and 20% of the data for the test portion.

	count	hate_spee	offensive_	neither	class	tweet
2	0	3	0	0	3	2 !!! RT @mayasolovely: As a woman you shouldn't complain about cleaning up your house. & as a man you should always take the trash out...
3	1	3	0	3	0	1 !!!!! RT @mleew17: boy dats cold...tyga dwn bad for cuffin dat hoe in the 1st place!!
4	2	3	0	3	0	1 !!!!!!! RT @UrKindOfBrand Dawg!!!! RT @80sbaby4life: You ever fuck a bitch and she start to cry? You be confused as shit
5	3	3	0	2	1	1 !!!!!!!!! RT @C_G_Anderson: @viva_based she look like a tranny
6	4	6	0	6	0	1 !!!!!!!!!!!! RT @ShenikaRoberts: The shit you hear about me might be true or it might be faker than the bitch who told it to ya 
7	5	3	1	2	0	1 !!!!!!!!!!!!!!!!!"@T_Madison_x: The shit just blows me..claim you so faithful and down for somebody but still fucking with hoes! 😂&#128514
8	6	3	0	3	0	1 !!!!!!"@__BrighterDays: I can not just sit up and HATE on another bitch .. I got too much shit going on!"
9	7	3	0	3	0	1 !!!!“@selfiequeenbri: cause I'm tired of you big bitches coming for us skinny girls!!”
10	8	3	0	3	0	1 " & you might not get ya bitch back & thats that "
11	9	3	1	2	0	1 "
12	10	3	0	3	0	1 " Keeks is a bitch she curves everyone " lol I walked into a conversation like this. Smh
13	11	3	0	3	0	1 " Murda Gang bitch its Gang Land "
14	12	3	0	2	1	1 " So hoes that smoke are losers ? " yea ... go on IG
15	13	3	0	3	0	1 " bad bitches is the only thing that i like "
16	14	3	1	2	0	1 " bitch get up off me "
17	15	3	0	3	0	1 " bitch nigga miss me with it "
18	16	3	0	3	0	1 " bitch plz whatever "

Figure 3.1: Dataset

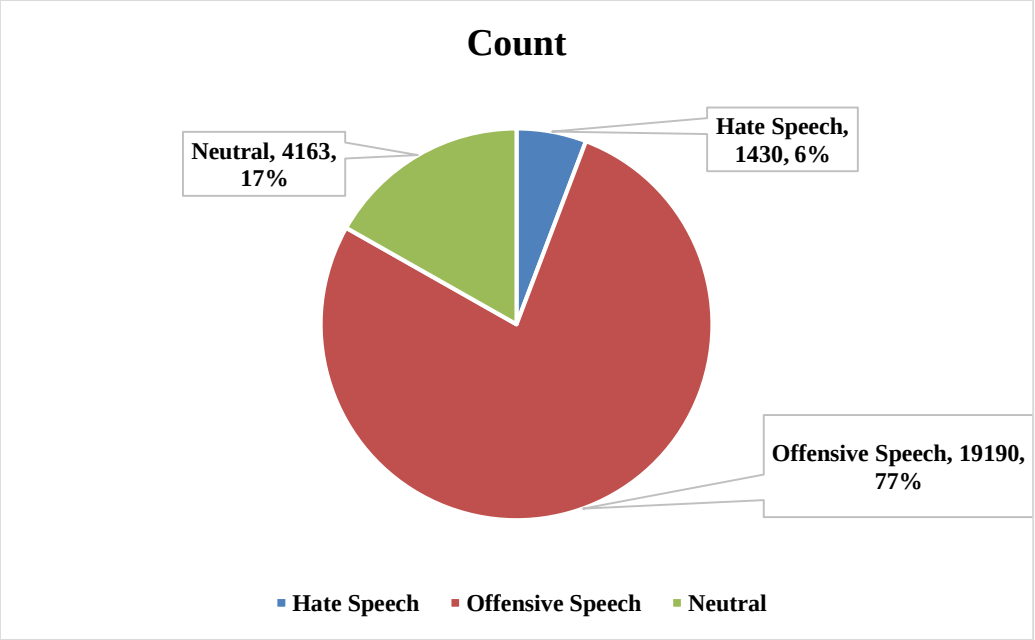


Figure 3.2: Number of target values dataset

There were no erroneous or missing values in the dataset, not even in the category values. Please refer to Table 3.1 for a thorough explanation of the dataset's contents and for an overview of its features.

Table 3.1: Details of the dataset

Columns	Description	Count
tweet	Contains text values	24783
class	Contains categorical values	24783

3.2.1 Categorical Data Encoding

category encoding is the process of translating numerical values corresponding to category variables. Considering that machine learning algorithms demand numerical data as input and output by nature, this encoding technique was critical to our study. In order to properly

implement this category encoding technique, we focused on the "class" column in our dataset.

3.2.2 Missing Value Imputation

In this process, partial or missing data that has been found via the examination of data from other datasets is replaced with imputed values. It is important to highlight that our dataset had the benefit of having no missing values, which removed the need for this kind of imputation.

3.2.3 Handling Imbalanced Data

By methodically adding new instances, this strategy may be used to manipulate the class distribution within a dataset and achieve effective data balancing. The main objective is to improve minority data representation using the complete dataset as input. We have employed SMOTE technique to balance the dataset class.

3.2.4 Remove Duplicate Rows

To delete duplicate rows from a dataset, one typically identifies and removes rows with identical values across all columns, ensuring each remaining row is unique. This process often involves using methods from data manipulation libraries like Pandas in Python, where the `drop_duplicates()` function can efficiently eliminate duplicates. This is crucial in data preprocessing to maintain data integrity, prevent redundancy, and ensure the accuracy of analyses and machine learning models. Removing duplicates helps in reducing the dataset size, enhancing performance, and preventing the bias that duplicate data might introduce into the analytical processes.

3.3 Statistical Analysis

In every research project, a critical component is the analysis section, which revolves around the development and evaluation of the employed algorithms. In our case, given the choice of working with an Excel file format, several preparatory steps were essential to ensure the dataset's usability. These measures included data collection and meticulous pre-

processing, both integral to the successful execution of our research. In this study, ten different types of algorithms were employed, namely Bagging, Adaboost, Gaussian Naïve Bayes, MLP, Random Forest, Decision Tree, XGBoost, KN Neighbors, Logistic Regression, and Voting. The Random Forest achieved an impressive accuracy of 96.04%.

3.4 Proposed Methodology

In our research, we employed a diverse array of classifiers, encompassing Bagging, Adaboost, Gaussian Naïve Bayes, MLP, Random Forest, Decision Tree, XGBoost, KN Neighbors, Logistic Regression, and Voting algorithms [20-30]. These classifiers played a crucial role in our data analysis and the comprehensive evaluation of algorithms.

Logistic Regression

Using one or more predictor variables to model the likelihood of a binary result, logistic regression is a statistical technique for binary classification. Typically coded as 0 or 1, logistic regression predicts the chance that an instance belongs to a certain category, as opposed to linear regression's prediction of a continuous result. This is achieved by applying the logistic function, often known as the sigmoid function, to translate the linear sum of the supplied input variables into an amount between 0 and 1. The model provides coefficients that indicate the influence of each predictor and calculates the likelihood that the dependent event will occur. In order to predict the existence of diseases, consumer behavior, and other things, logistic regression is often utilized in the social sciences, medical, and machine learning domains.

Random Forest

For problems involving regression and classification, an ensemble learning technique called a random forest is commonly employedIt works by creating a large number of decision trees during the training phase, from which it derives the average prediction (regression) or mode of grouping (classification) for every single tree. The "forest" is constructed using a process known as bagging (Bootstrap Aggregating), in which a decision tree is taught on each of the dataset's many subsets that are produced via

replacement. By averaging the variances and biases from each tree, this technique reduces overfitting and improves the generalization of the model. The usefulness of random forests in producing precise forecasts without requiring substantial parameter adjustment is well-known, as is their robustness and capacity to handle big datasets with increased dimensionality.

MLP

An artificial neural network with several layers of neurons, usually an input layer, one or more hidden layers, and an output layer, is called a multi-layer perceptron (MLP). An MLP becomes a completely linked network because every neuron in the next layer is connected to every other neuron in the MLP. ReLU and sigmoid are examples of nonlinear activation functions that MLPs utilize to induce nonlinearity, allowing the network to simulate complicated interactions between inputs and outputs. Back propagation is a supervised learning approach that is used to train them. It modifies weights according to the inaccuracy of the output compared to the desired outcome. MLPs can estimate any continuous function given enough data and computer power, which makes them popular in a variety of applications such as pattern recognition, regression, and classification.

Decision Tree

Regression and classification tasks are both suitable for the flexible and comprehensible machine learning model known as a decision tree. It creates a tree-like structure of decisions by iteratively dividing the dataset into subgroups according to feature values. A feature test is represented by each internal node, a test result is represented by each branch, and a class label or continuous value is represented by each leaf node. In order to create the tree, features and thresholds that optimally divide the data based on a selected criterion—such as mean squared error for regression and Gini impurity or information gain for classification—must be chosen. Decision trees are helpful for identifying correlations in the data because they are easy to comprehend and visualize, but if they are not regularly trimmed or adjusted, they may become overfit.

K-Nearest

The k-NN classifier is a simple, non-parametric, and easy-to-use method for machine learning classification and regression issues. In order to predict the output, it first locates the k instances in the training dataset that are most similar to a given input (neighbors). For regression, this is done by calculating the average value of these neighbors, and for classification, it uses the majority class. Similarity is frequently measured using distance metrics like Euclidean distance. Despite being simple to understand, k-NN can be useful, especially for small datasets or when the decision boundary is very irregular. However, it is computationally expensive for large datasets and dependent on the choice of k and the distance measure. Proper feature selection and data scaling are necessary for improved data performance.

XGBoost

Developed especially for supervised learning tasks, the XGBoost (Extreme Gradient Boosting) classifier is a potent and effective application of the gradient boosting architecture. It achieves superior predicted performance by building a strong predictive model from an ensemble of weak prediction models, usually decision trees. XGBoost offers several key advantages, including handling missing values, incorporating regularization to prevent overfitting, and leveraging advanced tree learning algorithms that optimize speed and performance. Its scalability and flexibility make it a popular choice for various applications, from structured/tabular data to more complex domains, offering superior accuracy and efficiency in comparison to many traditional machine learning algorithms.

Gaussian Naïve Bayes

One of the fundamental techniques in machine learning is the Naive Bayes algorithm, which is well-known for its ease of use, computational efficacy, and versatility in handling different categorization tasks. The core of it is the Bayes theorem, a basic idea in probability theory that determines the likelihood of a hypothesis based on observable data. Based on the observed attributes, Naive Bayes determines the likelihood that a given data point belongs to a specific class in the context of classification. What distinguishes Naive

Bayes from other classifiers is its "naive" assumption of feature independence, which simplifies the computation of probabilities by assuming that each feature contributes independently to the probability of the data point belonging to a certain class. This assumption, while simplistic, often holds up well in practice, especially in text classification tasks and other domains where feature dependencies are not significant. The simplicity of Naive Bayes stems from its straightforward approach to modeling. Rather than learning complex relationships between features, Naive Bayes directly estimates probabilities from the training data. This simplicity makes Naive Bayes particularly well-suited for situations with limited training data, where more complex models may struggle due to overfitting. Moreover, Naive Bayes classifiers are computationally efficient, making them ideal for large datasets and high-dimensional feature spaces. The algorithm's efficiency arises from its ability to compute probabilities independently for each feature, resulting in a linear time complexity with respect to the number of features. There are several variations of naive Bayes classifiers, each designed to handle distinct kinds of data. For classification tasks involving continuous-valued data, the Gaussian Naive Bayes version is appropriate since it makes the assumption that continuous features have a Gaussian (normal) distribution. This variation is frequently employed in settings where characteristics like patient age or blood pressure may have a normal distribution, such as in medical diagnosis applications. The Multinomial Naive Bayes variation, on the other hand, is made for text classification problems, where features are words or keywords that appear often in texts. In this case, the algorithm assumes that features follow a multinomial distribution, making it well-suited for tasks such as sentiment analysis, spam detection, and document categorization. Additionally, the Bernoulli Naive Bayes variant is similar to the Multinomial variant but is specifically tailored for binary feature vectors, where features represent the presence or absence of terms in documents.

Bagging

Bootstrap Aggregating, or Bagging, is a strategy for ensemble learning that aims to increase the precision and resilience of machine learning models. It involves training several base models, often decision trees, utilizing multiple subsets of the training data generated using bootstrap sampling, or random sampling with replacement. Every model

undergoes separate training, and for classification tasks, the predictions are aggregated by a majority vote, and for regression tasks, through averaging. Bagging, as opposed to using a single model, produces more consistent and dependable performance by combining the predictions of many models, which lowers variance and helps prevent overfitting. One of the most well-known uses of this technique is Random Forest, which expands bagging by adding more randomization to the characteristics chosen for each split in the trees.

Adaboost

An effective machine learning method that builds a strong overall model by combining several weak classifiers is called a boosting ensemble classifier. Boosting is based on the sequential training of classifiers, where each new model learns from the mistakes made by the prior ones. This is usually accomplished by giving misclassified cases larger weights, which incentivizes the subsequent classifier in the series to fix the errors. AdaBoost and Gradient Boosting are two popular boosting methods that iteratively modify the weights of the classifiers and training data. Boosting improves the final classifier's accuracy and resilience by combining the predictions of all the models, frequently beating the performance of individual models, especially in situations with complicated patterns and noisy data.

Voting

A machine learning model called a voting ensemble classifier aggregates the predictions of several different classifiers to increase overall resilience and accuracy. The way the ensemble functions is by using a voting mechanism, which can be either soft or hard voting, to aggregate the predictions of its component models. Hard voting selects the class with the majority of votes as the final forecast, with each classifier voting for a particular class label. During the soft voting process, the class with the greatest average probability is selected by averaging the projected probabilities of each class from all classifiers. This approach leverages the strengths of diverse models, reducing the likelihood of errors and increasing the performance, especially in complex and noisy datasets. By pooling the decisions of several models, a voting ensemble can achieve better generalization compared to any individual model alone.

Proposed Methodology Flow chart:

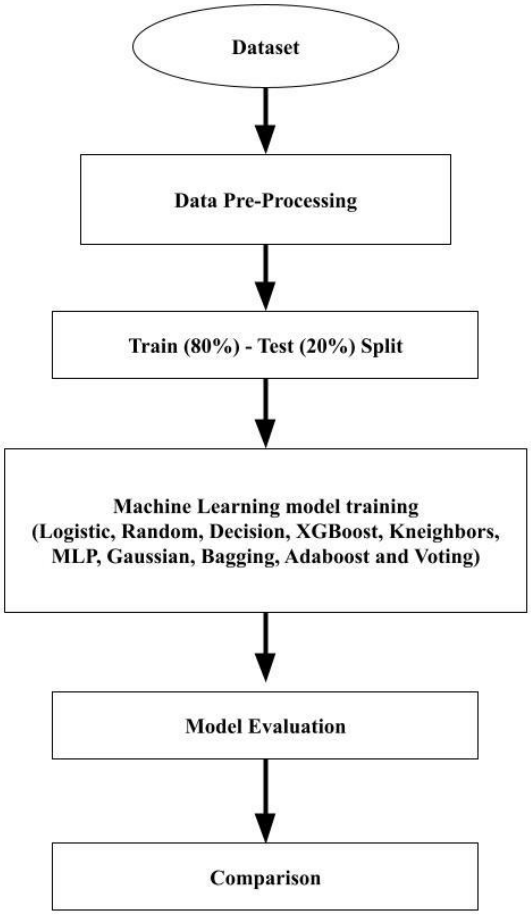


Figure 3.3: Methodology

Several important phases are included in the offensive detection process that was applied to the English dataset. First, a dataset of English social media comments is collected. Next, preparatory operations such deleting unnecessary columns, dealing with missing values, and text cleaning are performed. The dataset is then divided into training and testing sets with an 80-20 ratio, and the target column is numerically mapped to aid in model training [18]. Numerical values are created from categorical data in order to facilitate further analysis. To prepare text data for input into a variety of models, such as Bagging, Adaboost, Gaussian Naïve Bayes, MLP, Random Forest, Decision Tree, XGBoost, KN Neighbors, Logistic Regression, and Voting, it is tokenized and padded. The prepared data is used to

train these models. Evaluation metrics are used to evaluate the performance of the models, and for models using probability estimates, these measures include accuracy, precision, recall, and F1 score. Choosing the top-performing model is the last stage, which is determined by the evaluation's findings [19]. For researchers looking to create efficient offensive detection algorithms, this thorough technique gives a solid framework for identifying offensive in the English dataset [20].

3.5 Implementation Requirements

Acquiring the required datasets was the first step in our process to guarantee the efficacy of our suggested model for offensive detection. A rigorous data cleaning procedure was put in place, using a variety of filtering techniques to ensure the integrity of the dataset. Important data pretreatment operations were then carried out, such as using the Scaler Transform to transform categorical data into numerical values. After that, the dataset was divided in an 80-20 ratio into training and testing sets, giving a strong basis for evaluating algorithms. The selected methods were put into practice, and the outcomes were thoroughly evaluated. To improve the detection accuracy, ensemble methods were used, and the results of these ensemble algorithms were thoroughly verified by hyperparameter tweaking, which required a careful analysis of the underlying mathematical models. Then came the data analysis stage, which included creating a personalized detection approach and learning a model. On the basis of important measures including accuracy, precision, recall, and the F-1 score, the top-performing model was chosen.

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Experimental Setup

The methodology used in this work is based on supervised learning and consists of distinct training and assessment phases. The testing dataset was used to extensively assess the model's performance, while the training dataset was utilized to begin developing the machine learning model. The machine learning technique employed in this work will be briefly but thoroughly explained in the sections that follow.

4.2 Experimental Results & Analysis

At this point, we shifted our attention to analyzing the performance of the models that already exist as well as the effectiveness of our suggested model in the field of offensive detection. We methodically used a variety of performance evaluation techniques using never-before-seen data to assess overall performance. An analytical report based on our experimental results—more particularly, machine learning models customized for offensive datasets—is summarized in this section. The first step in our process was to apply our selected dataset, making sure that any missing or inaccurate numbers were carefully handled to maintain the dataset's integrity. Then, a wide variety of algorithms were presented, and their performance was thoroughly examined. Important measures including the F-1 Score, Accuracy, Precision, Recall, and Confusion Matrices were used to assess these algorithms' efficacy in-depth. Both the conventional techniques and our suggested models were evaluated. Several methods, customized for our dataset, were used in our analysis: Bagging, Adaboost, Gaussian Naïve Bayes, MLP, Random Forest, Decision Tree, XGBoost, KN Neighbors, Logistic Regression, and Voting. Several ensemble strategies were investigated to improve our evaluation even further, and Confusion Matrices offered insights for both datasets. The goal of this thorough approach was to provide a full knowledge of the efficacy and performance of the various models that were being considered in the context of offensive detection.

Accuracy

This section explores the idea of accuracy, which is the proportion of predictions produced with testing data that turned out to be accurate or precise. By comparing the model's predictions to actual measurements made in the real world, accuracy serves as a gauge for the model's correctness. It is one of the simplest and most popular model assessment methods since it concentrates on just one variable and mostly deals with deliberate mistakes. A critical component of model validation and performance evaluation is ensuring the correctness of our models.

$$Accuracy = \frac{TruePositive + TrueNegative}{TruePositive + FalsePositive + TrueNegative + FalseNegative}$$

Precision

Precision is the measure of the percentage of favorably predicted observations that really happened, and it is covered in this section. The real positive rate is reflected in precision, which shows the actual percentage of cases in which the model accurately predicted genuine positive outcomes. It's crucial to remember that, even though many models prefer a high recall, this characteristic can occasionally be deceptive if accuracy and other performance measures aren't taken into account.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

Recall

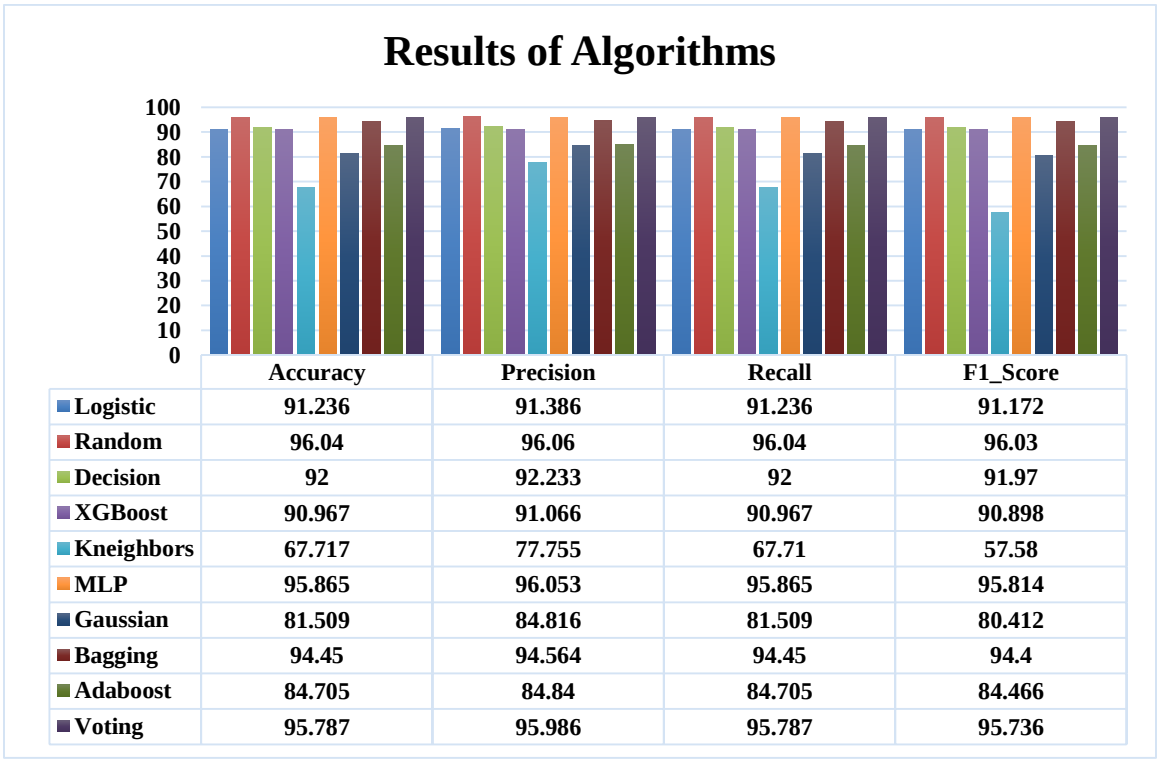
Recall, or the percentage of real positive data items that the model properly predicts, is covered in this section. Recall determines the ratio of all positive labels to the expected positives and is important in assessing the model's capacity to catch genuine positive occurrences. High accuracy is usually preferred, but it's vital to understand that it can occasionally be deceptive if evaluated in isolation from other crucial measures, such as recall.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

F-1 Score

The assessment metrics of accuracy and recall are covered in this part, with a focus on their use in evaluating a model's performance. The recall and accuracy ratios are important metrics to take into account since they provide light on the model's overall accuracy and correctness in identifying pertinent occurrences. It is noteworthy that a comparatively low mean of the harmonic mean of these measures may suggest suboptimal performance of the model, necessitating more refinements.

$$F - 1\ Score = 2 * \frac{Recall * Precision}{Recall + Precision}$$



	Accuracy	Precision	Recall	F1_Score
Logistic	91.236	91.386	91.236	91.172
Random	96.04	96.06	96.04	96.03
Decision	92	92.233	92	91.97
XGBoost	90.967	91.066	90.967	90.898
Kneighbors	67.717	77.755	67.71	57.58
MLP	95.865	96.053	95.865	95.814
Gaussian	81.509	84.816	81.509	80.412
Bagging	94.45	94.564	94.45	94.4
Adaboost	84.705	84.84	84.705	84.466
Voting	95.787	95.986	95.787	95.736

Figure 4.1: Experimental Results

Analysis of Each Algorithm

Table 4.1 Result analysis of LR

Algorithm	Accuracy	Precision	Recall	F1 Score
LR	**91.236**	**91.386**	**91.236**	**91.172**

Logistic Regression, a simple yet powerful linear model, shows strong performance with an accuracy of 91.236%. Its precision (91.386%) and recall (91.236%) are nearly identical, indicating a balanced model with minimal false positives and false negatives. The F1 score (91.172%) confirms this balance. Logistic Regression is often preferred for its interpretability and efficiency in binary classification tasks.

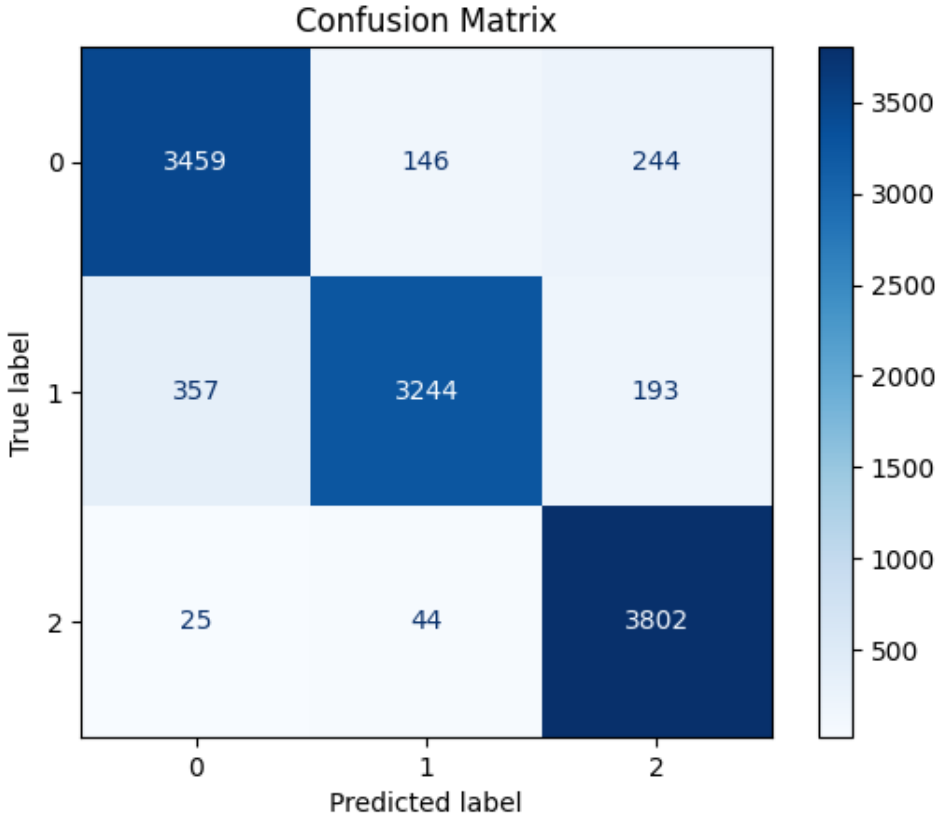


Figure 4.2: Confusion Matrix of Logistic Regression

Random Forest

Table 4.2 Result analysis of RF

Algorithm	Accuracy	Precision	Recall	F1 Score
RF	**96.04**	**96.06**	**96.04**	**96.03**

Random Forest, an ensemble learning method, demonstrates outstanding performance with an accuracy of 96.04%. This algorithm combines the predictions of multiple decision trees, resulting in high precision (96.06%) and recall (96.04%). The slight differences between these metrics and the F1 score (96.03%) suggest that Random Forest manages the bias-variance trade-off effectively, providing robust predictions and reducing the impact of overfitting.

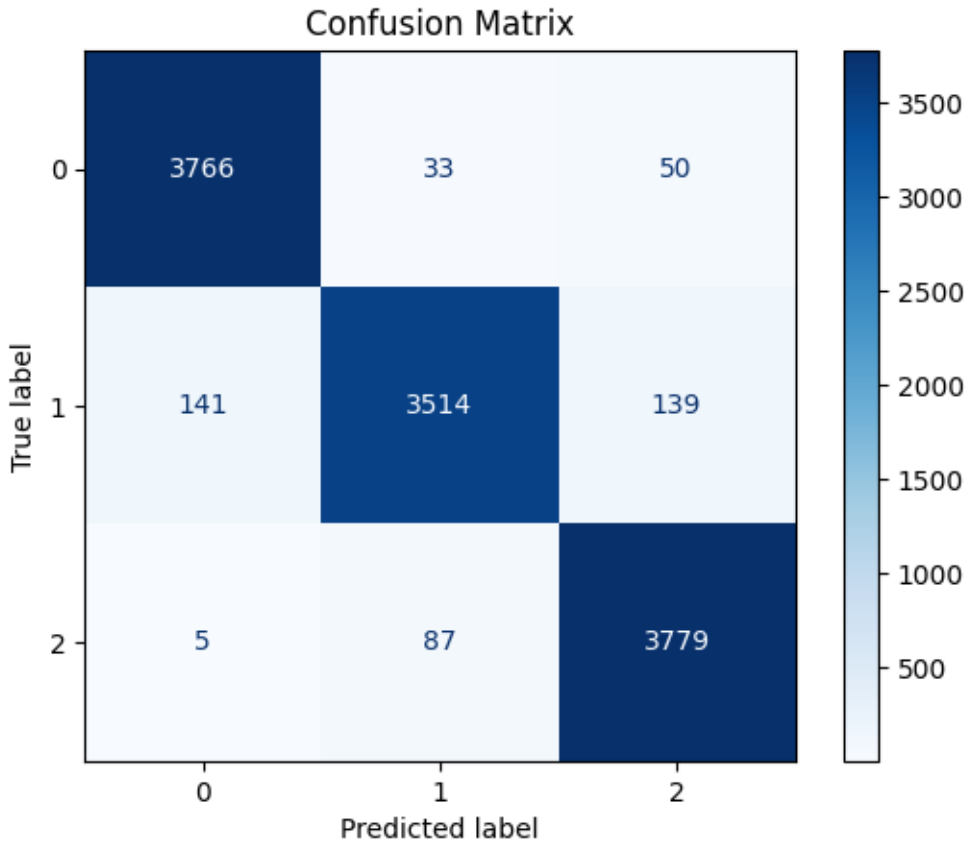


Figure 4.3: Confusion Matrix of Random Forest

Decision Tree

Table 4.3 Result analysis of DT

Algorithm	Accuracy	Precision	Recall	F1 Score
DT	**92**	**92.233**	**92**	**91.97**

Decision Trees, known for their simplicity and interpretability, achieve an accuracy of 92%. The precision (92.233%) is slightly higher than the recall (92%), indicating a marginally lower rate of false positives compared to false negatives. The F1 score (91.97%) supports this observation. While Decision Trees can capture complex decision boundaries, they are prone to overfitting, which might explain the slightly lower performance compared to Random Forest.

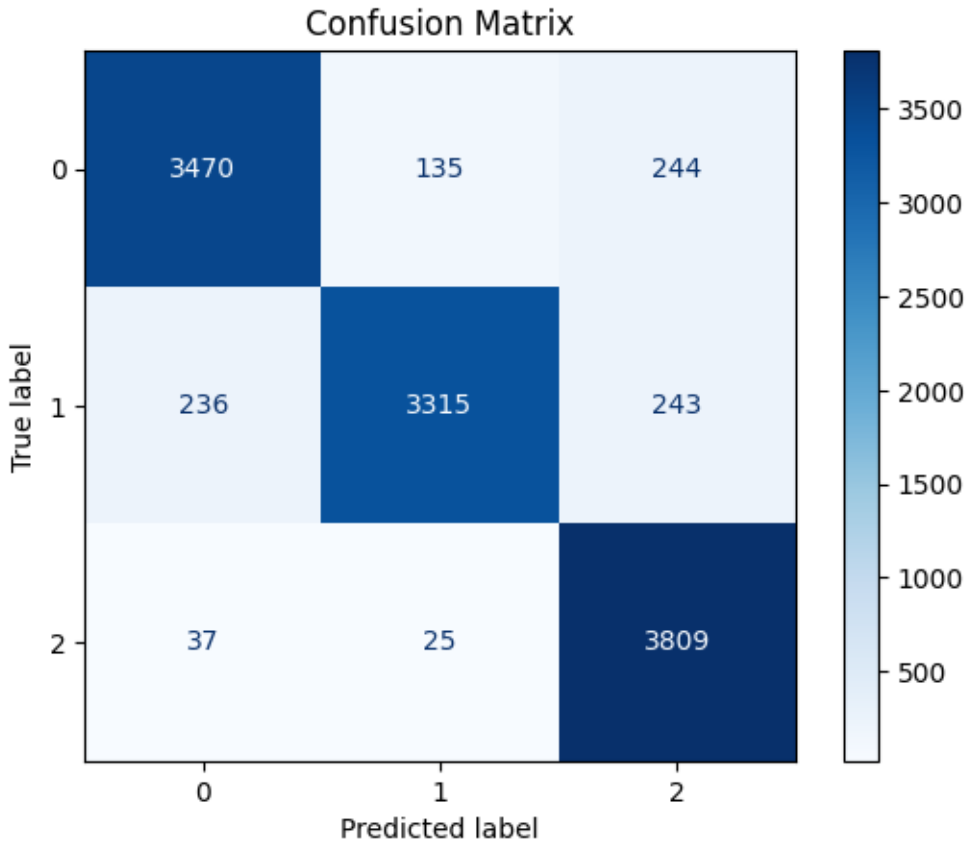


Figure 4.4: Confusion Matrix of Decision Tree

XGBoost

Table 4.4 Result analysis of XGBoost

Algorithm	Accuracy	Precision	Recall	F1 Score
XGBoost	**90.967**	**91.066**	**90.967**	**90.898**

XGBoost, a powerful gradient boosting algorithm, exhibits strong performance with an accuracy of 90.967%. Its precision (91.066%) is marginally higher than its recall (90.967%), suggesting effective identification of positive instances with a slightly higher false positive rate. The F1 score (90.898%) reflects a well-balanced model. XGBoost is known for its efficiency and performance, especially in structured datasets, and is often a go-to choice in machine learning competitions.

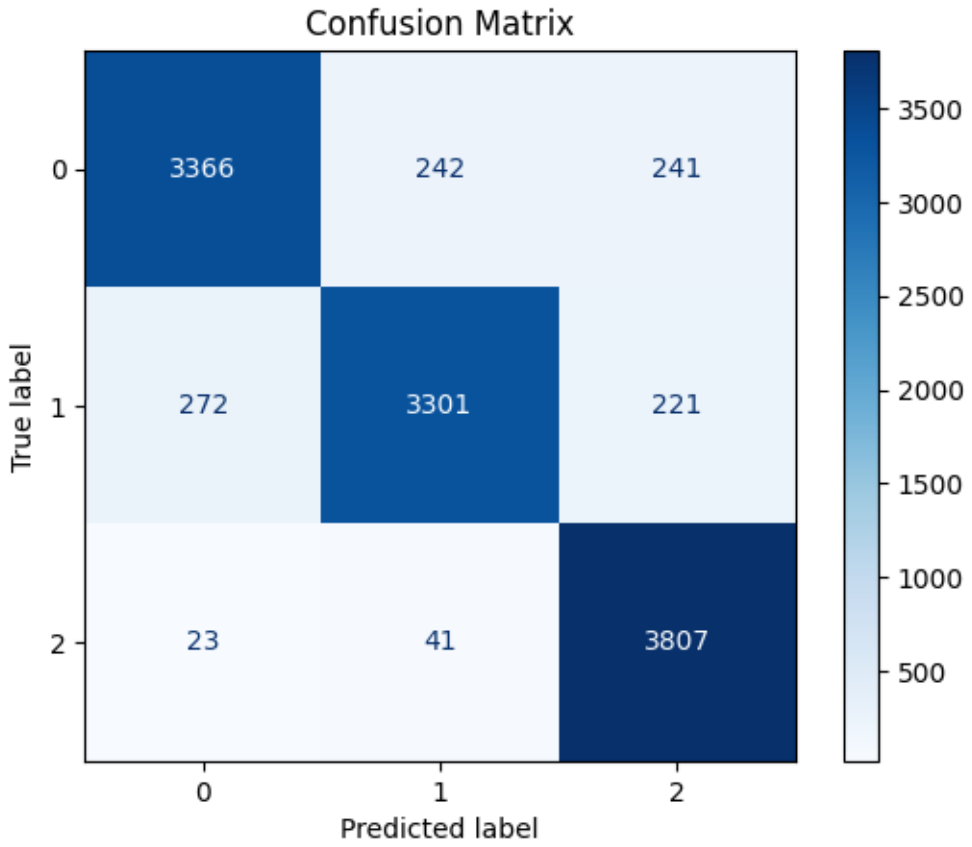


Figure 4.5: Confusion Matrix of XGBoost

K-Nearest Neighbors (KNN)

Table 4.5 Result analysis of KNN

Algorithm	Accuracy	Precision	Recall	F1 Score
KNN	**67.717**	**77.755**	**67.71**	**57.58**

KNN shows relatively lower performance, with an accuracy of 67.717%. The high precision (77.755%) indicates that the model has a significant number of false negatives, as evidenced by the lower recall (67.71%). The F1 score (57.58%) is substantially lower, highlighting the trade-off between precision and recall. KNN is a simple, instance-based learning algorithm that can be sensitive to noisy data and outliers, which might explain its lower performance in this evaluation.

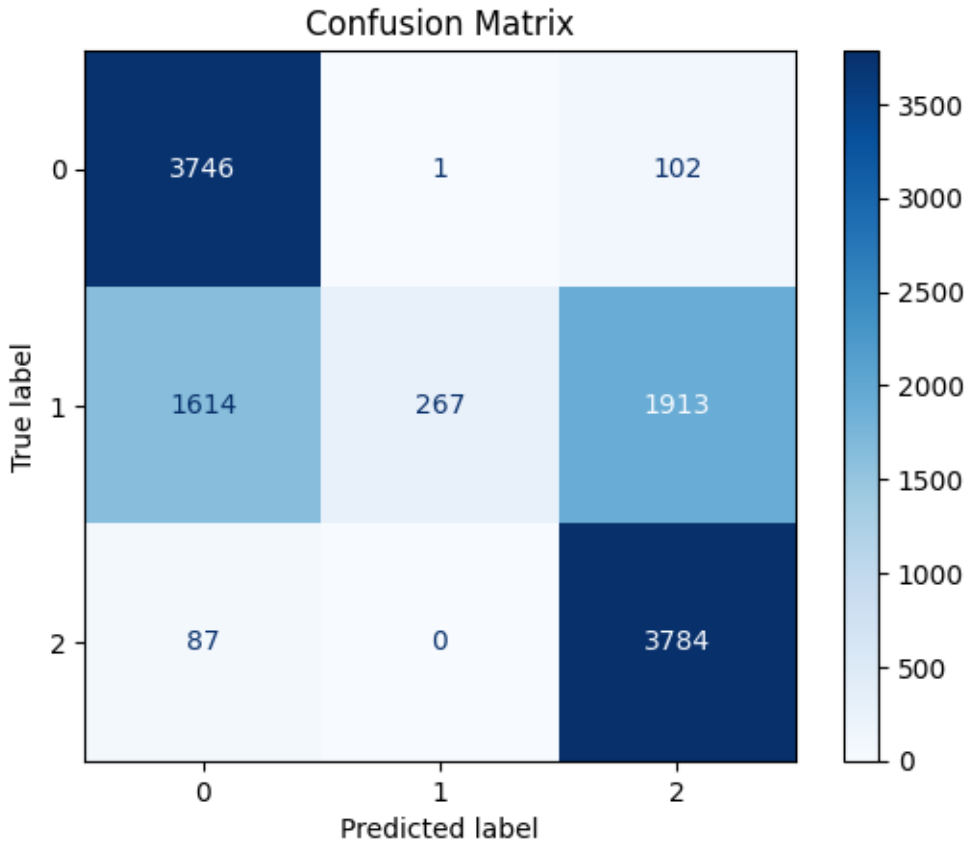


Figure 4.6: Confusion Matrix of K-Nearest Neighbors (KNN)

Multi-Layer Perceptron (MLP)

Table 4.6 Result analysis of MLP

Algorithm	Accuracy	Precision	Recall	F1 Score
MLP	**95.865**	**96.053**	**95.865**	**95.814**

MLP, a type of neural network, demonstrates excellent performance with an accuracy of 95.865%. Its precision (96.053%) and recall (95.865%) are closely aligned, suggesting a balanced model with minimal false positives and false negatives. The F1 score (95.814%) confirms this balance. MLPs are capable of capturing complex patterns in data, making them suitable for a wide range of tasks, including classification and regression.

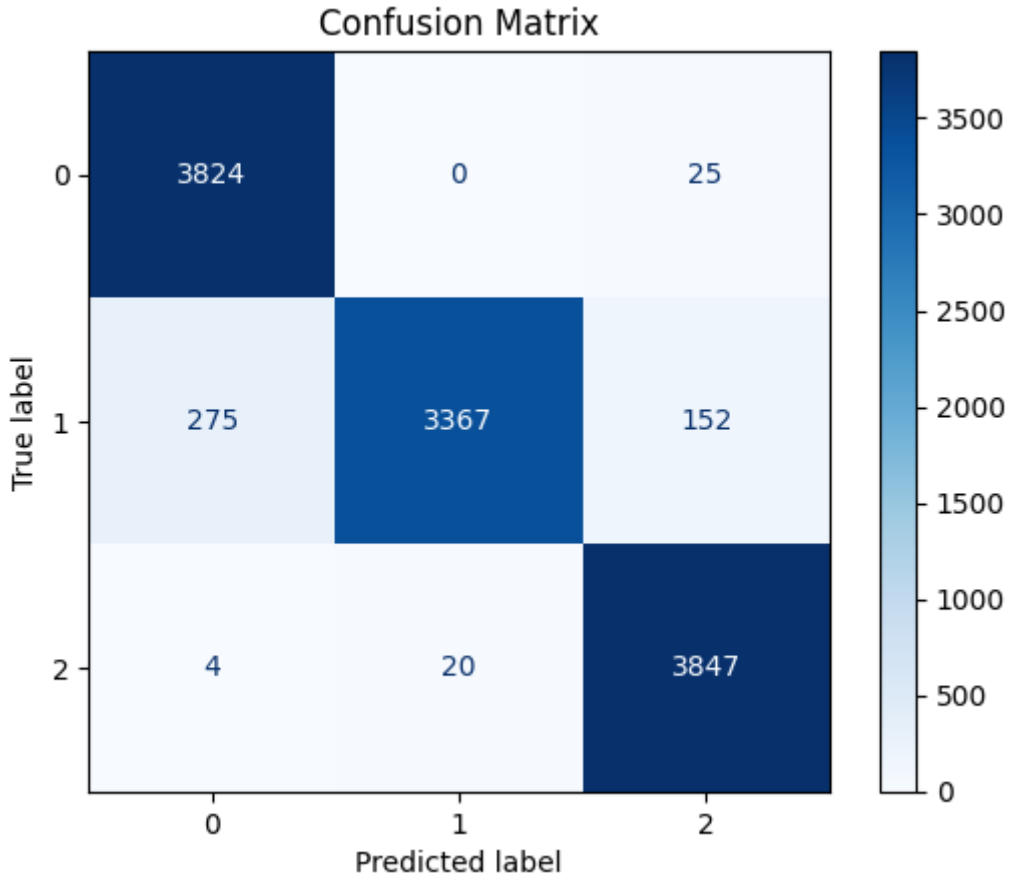


Figure 4.7: Confusion Matrix of Multi-Layer Perceptron (MLP)

Gaussian Naive Bayes

Table 4.7 Result analysis of GNB

Algorithm	Accuracy	Precision	Recall	F1 Score
GNB	**81.509**	**84.816**	**81.509**	**80.412**

Gaussian Naive Bayes shows moderate performance, with an accuracy of 81.509%. The precision (84.816%) is higher than the recall (81.509%), indicating a higher rate of false negatives. The F1 score (80.412%) highlights this imbalance. Naive Bayes is a probabilistic classifier based on Bayes' theorem, assuming independence between features. Its simplicity and efficiency make it a good choice for text classification and other tasks with a strong feature independence assumption.

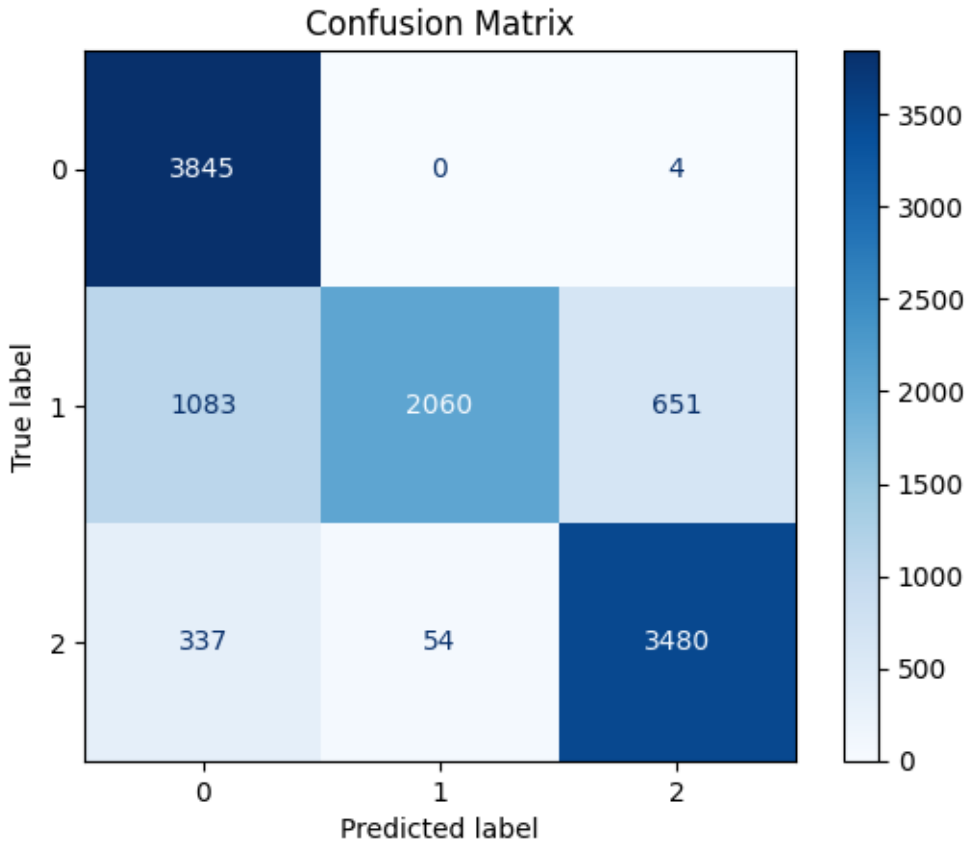


Figure 4.8: Confusion Matrix of Gaussian Naive Bayes

Bagging Classifier

Table 4.8 Result analysis of Bagging

Algorithm	Accuracy	Precision	Recall	F1 Score
Bagging	**94.45**	**94.564**	**94.45**	**94.4**

Bagging, or Bootstrap Aggregating, shows strong performance with an accuracy of 94.45%. Its precision (94.564%) and recall (94.45%) are nearly identical, indicating a balanced model with minimal false positives and false negatives. The F1 score (94.4%) confirms this balance. Bagging reduces variance by averaging multiple models trained on different subsets of the data, making it robust to overfitting and improving stability.

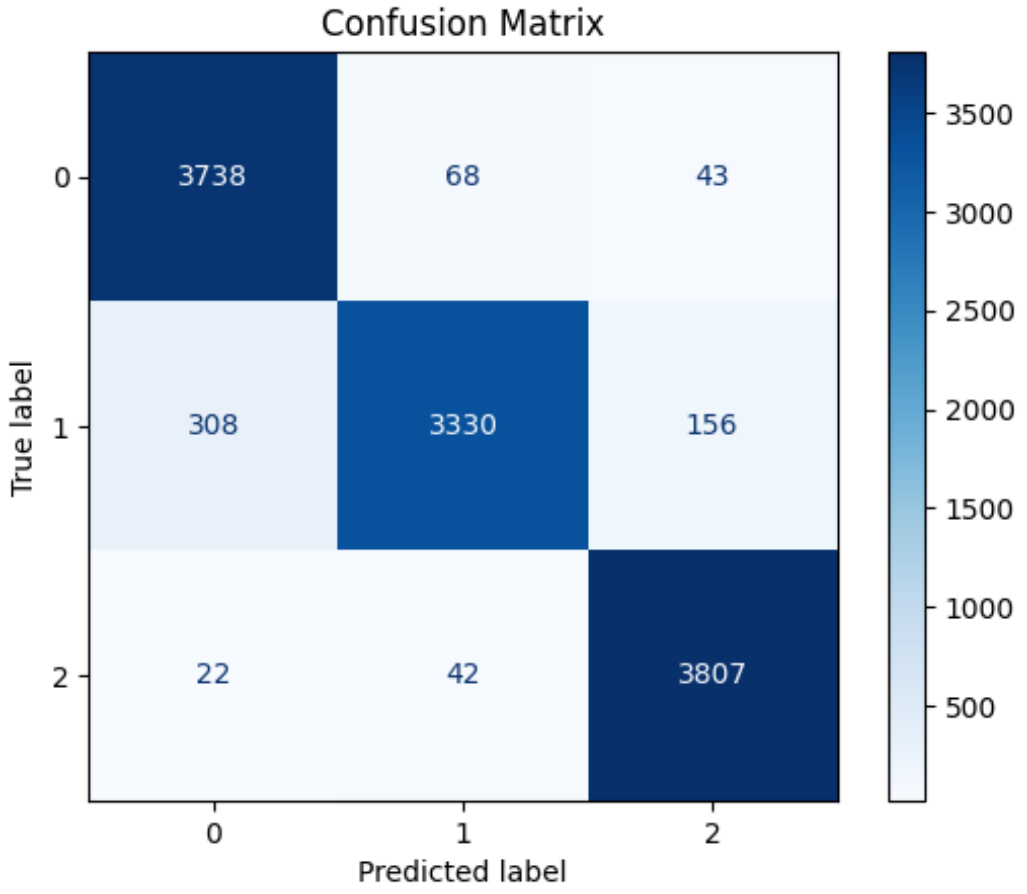


Figure 4.9: Confusion Matrix of Bagging Classifier

AdaBoost

Table 4.9 Result analysis of AdaBoost

Algorithm	Accuracy	Precision	Recall	F1 Score
AdaBoost	**84.705**	**84.84**	**84.705**	**84.466**

AdaBoost, a boosting algorithm, demonstrates moderate performance with an accuracy of 84.705%. Its precision (84.84%) and recall (84.705%) are closely aligned, suggesting a balanced model with minimal false positives and false negatives. The F1 score (84.466%) supports this balance. AdaBoost works by combining weak learners to form a strong classifier, focusing on misclassified instances in subsequent iterations, which enhances its ability to handle complex datasets.

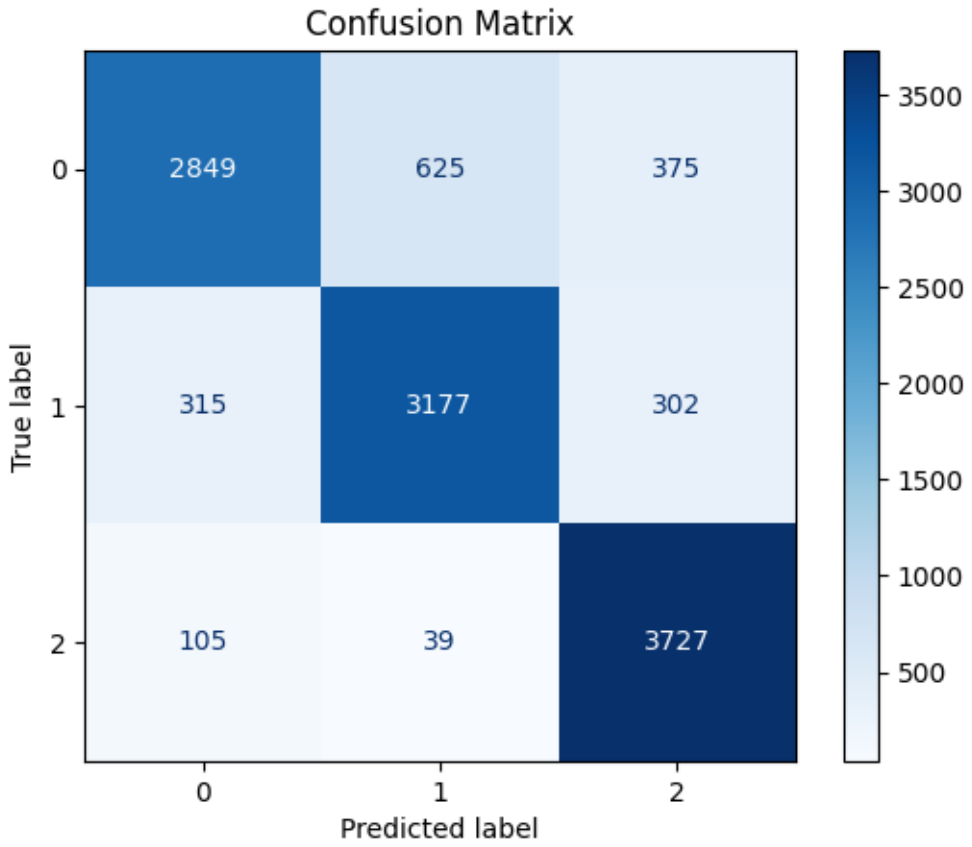


Figure 4.10: Confusion Matrix of AdaBoost

Voting Classifier

Table 4.10 Result analysis of Voting

Algorithm	Accuracy	Precision	Recall	F1 Score
Voting	**95.787**	**95.986**	**95.787**	**95.736**

The Voting Classifier, an ensemble method that combines the predictions of multiple models, shows excellent performance with an accuracy of 95.787%. Its precision (95.986%) and recall (95.787%) are closely aligned, indicating a balanced model with minimal false positives and false negatives. The F1 score (95.736%) confirms this balance. The Voting Classifier leverages the strengths of different models, often leading to improved performance and robustness.

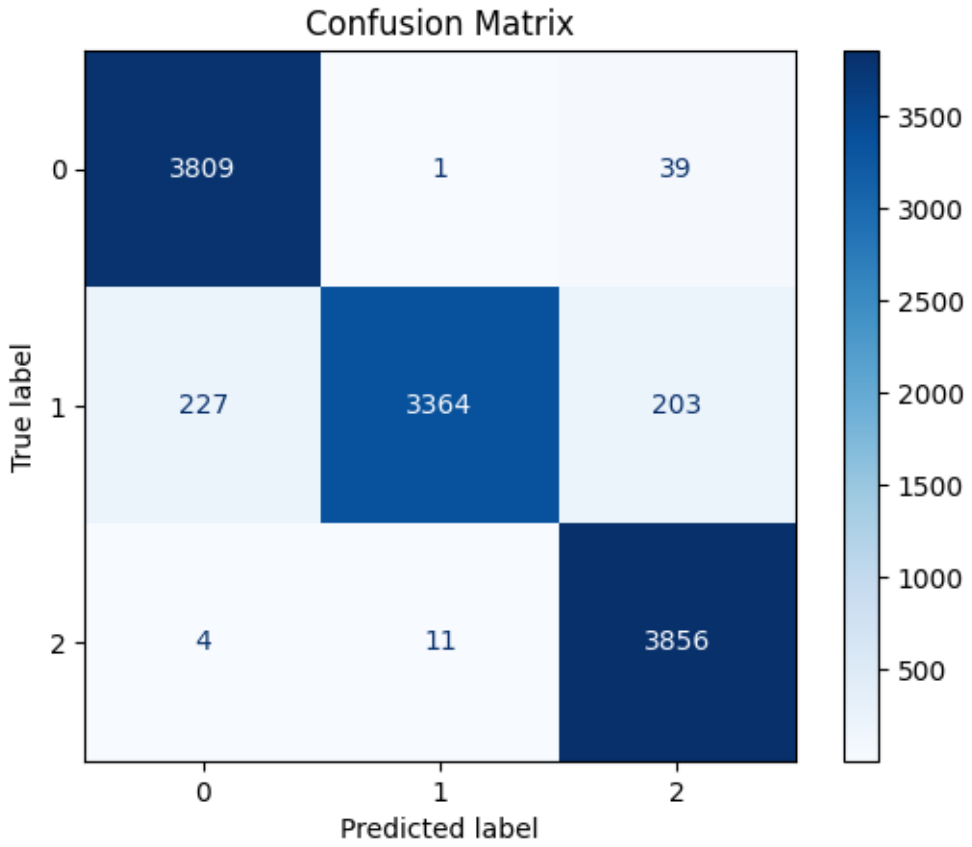


Figure 4.11: Confusion Matrix of Voting Classifier

4.3 Discussion

We shall clarify the evaluation framework for our suggested model at this point. We used accuracy, precision, recall, and the F-1 score as our evaluation criterion [22]. The evaluation results provide valuable insights into the strengths and weaknesses of each algorithm. Here's a comparative analysis based on the key metrics:

Accuracy: Random Forest (96.04%) and MLP (95.865%) show the highest accuracy, indicating their strong overall performance. KNN (67.717%) has the lowest accuracy, reflecting its sensitivity to noisy data and the curse of dimensionality.

Precision: Random Forest (96.06%) and MLP (96.053%) have the highest precision, suggesting their effectiveness in minimizing false positives. KNN (77.755%) has a relatively high precision compared to its accuracy, indicating a higher false negative rate.

Recall: Random Forest (96.04%) and MLP (95.865%) also excel in recall, indicating their ability to identify positive instances effectively. KNN (67.71%) has the lowest recall, reflecting its higher rate of false negatives.

F1 Score: The F1 scores of Random Forest (96.03%) and MLP (95.814%) are the highest, indicating their balanced performance across precision and recall. KNN (57.58%) has the lowest F1 score, highlighting the trade-off between precision and recall.

Specificity: Specificity, or the true negative rate, is particularly high for algorithms like Random Forest and MLP, which perform well across all other metrics. This indicates their robustness in correctly identifying negative instances.

4.4 Limitation

We have considered used the open source dataset from online. We have assumed the data pre-processing for our needs. The pre-processing was suitable for our datasets. It may perform less in any other datasets.

CHAPTER 5

IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY

5.1 Impact on Society

The consequences of offensive behavior in the English-speaking community are extensive and impact several facets of people's life. Victims experience severe psychological effects, including elevated tension, anxiety, and depressive symptoms. The burden of attack also falls on academic endeavors and successes; sufferers frequently experience problems including increased absenteeism, decreased motivation, and trouble focusing. Social interactions suffer because people who are offended may choose to isolate themselves out of fear or embarrassment, which leaves them feeling isolated and excluded from society. In addition, victims' reputations and images are jeopardized, impacting their lives in both the personal and professional domains. Because abusive language in the English language is so widespread, it is critical to take proactive steps to lessen its effects and create a safer online environment.

5.2 Impact on Environment

It is essential to underscore that offensiveness mostly takes on digital form as opposed to bodily form. But it has significant effects on the internet environment. Offensive creates an environment that is hostile, fearful, and negative, which deters people from engaging in online discourse and freely expressing themselves. The pervasiveness of offensive content erodes trust, empathy, and respect for one another while undermining the development of a positive and welcoming online community. It is essential to support initiatives that address offensive content and foster a good online community in order to promote healthy digital interactions and improve the wellbeing of those using the internet.

5.3 Ethical Aspects

The identification of offensiveness in the English language raises important ethical issues. Ensuring the privacy and confidentiality of study participants is crucial in order to protect

sensitive data and personal information. Respecting the rights and permission of the people whose data is being used requires that ethical rules be followed while gathering and analyzing comments from social media. Maintaining impartiality and fairness is essential to avoiding prejudice and stigmatizing particular groups. In order to allow for examination and accountability, transparency in the deployment of algorithms and models is essential. In order to protect the rights and well-being of everyone affected by offensive, ethical awareness and responsible behavior are essential. This also advances knowledge of the problem and its prevention in English-speaking culture.

5.4 Sustainability Plan

Long-term strategies and actions are needed to guarantee long-lasting impact and efficacy when developing a sustainable strategy to offensive detection in the English language. In order to carry out awareness campaigns, policies, and support systems, this entails forming partnerships with important stakeholders including social media platforms, educational institutions, and community groups. In order to adjust to changing attack strategies, it is imperative that detection methods and algorithms be continuously monitored and evaluated. Over time, the accuracy and relevance of the detection model may be improved with regular updates and enhancements that take into account user input and technological breakthroughs. Furthermore, promoting a safer and healthier digital environment for the English-speaking population may be achieved by supporting a culture of digital empathy, resilience, and responsible online conduct through education and awareness initiatives.

CHAPTER 6

SUMMARY, CONCLUSION, RECOMMENDATION AND IMPLICATION FOR FUTURE RESEARCH

6.1 Summary of the Study

The goal of this project is to identify objectionable content using English-specific machine learning algorithms. With a focus on addressing the language and cultural peculiarities common in the English-speaking society, the goal is to close the research gap now present for offensive detection in English situations. Through the use of machine learning models—more especially, Random Forest cells—the research aims to develop a strong model that can recognize objectionable content in English—especially in comments on social media. The results of this study aim to substantially contribute to the creation of language-specific tools, policies, and treatments, taking into account the particular issues related to offensiveness in the English language, which include linguistic complexity and cultural considerations. These in turn seek to deter and mitigate offensive acts, creating a more secure and welcoming online space for people using English for communication.

6.2 Conclusions

To sum up, this work greatly advances the essential problem of offensive in English settings by detecting offensiveness in the English language using machine learning approaches. The creation of an English-specific machine learning model takes into consideration the language and cultural quirks that influence offensive behaviors in this particular group. The study emphasizes the significance of using language-specific strategies and the effectiveness of machine learning methods, particularly recurrent neural networks with Random Forest cells, in correctly detecting objectionable content in English text. The study emphasizes the need for inclusive techniques that cater to varied linguistic communities and fills a critical research gap in offensive detection for English languages. By focusing on the English language, the study hopes to empower those who use it for communication, guaranteeing fair attention and defense against abusive words. The

practical ramifications of the study's findings extend to the creation of language-specific policies, interventions, and instruments aimed at preventing and lessening offensive occurrences within the English-speaking population. The study highlights how crucial it is to continuously monitor, assess, and improve detection algorithms in order to adjust to changing attack strategies. It also emphasizes how important it is to use awareness campaigns and educational activities to foster a culture of digital empathy, resilience, and responsible online conduct. Essentially, the present study establishes the foundation for subsequent investigations and endeavors aimed at countering offensive language in English. It emphasizes how important it is for academics, legislators, social media companies, academic institutions, and community organizations to work together to create an online space that is safer and more welcoming for everyone, regardless of the language they speak.

6.3 Implication for Further Study

Subsequent investigations concerning English language offensive detection ought to give precedence to the improvement and enlargement of datasets so as to cover a wider range of offensive occurrences and environmental conditions. Adopting sophisticated machine learning models, especially transformer-based architectures such as GPT or BERT, can improve offensive detection systems' accuracy and effectiveness. One area of research that has to be looked into is model transferability, or how well models trained on bigger, multilingual datasets translate to English. Furthermore, longitudinal research can provide a more profound comprehension of the dynamic character of assault and its long-lasting effects on individuals throughout time, providing important information for the creation of preventative interventions and support networks. These approaches highlight the ongoing development and improvement of offensive detection techniques adapted to the subtle language and cultural differences within the English-speaking society.

Reference:

[1] “Hate Speech and Offensive Language Dataset”, Access: https://www.kaggle.com/datasets/mrmorj/hate-speech-and-offensive-language-dataset/data

[2] L. Yang and A. Shami, “On hyperparameter optimization of machine learning algorithms: theory and practice,” Neurocomputing, vol. 415, pp. 295–316, 2020.

[3] Malik, H., Shakshuki, E.M., et al.: Approximating viewership of streaming TVprograms using social media sentiment analysis. Procedia Comput. Sci. 198, 94–101 (2022)

[4] El Rahman, S.A., AlOtaibi, F.A., AlShehri, W.A.: Sentiment analysis of Twitterdata. In: 2019 International Conference on Computer and Information Sciences(ICCIS), pp. 1–4. IEEE (2019)

[5] Samghabadi, Niloofar Safi, et al.” Detecting nastiness in social media.” Proceedings of the First Workshop on Abusive Language Online. 2017.

[6] Yao, Mengfan, Charalampos Chelmis, and Daphney? Stavroula Zois.” Cyberbullying ends here: Towards robust detection of cyberbullying in social media.” The World Wide Web Conference.2019.

[7] Neogi, A.S., Garg, K.A., Mishra, R.K., Dwivedi, Y.K.: Sentiment analysis andclassification of Indian farmers’ protest using Twitter data. Int. J. Inform. Manage.Data Insights 1(2), 100019 (2021)

[8] Hasan, M.R., Maliha, M., Arifuzzaman, M.: Sentiment analysis with NLP on Twit-ter data. In: 2019 International Conference on Computer, Communication, Chem-ical, Materials and Electronic Engineering (IC4ME2), pp. 1–4. IEEE (2019)

[9] Huang, Qianjia, Vivek Kumar Singh, and Pradeep Kumar Atrey.” Cyberbullying detection using social and textual analysis.” Proceedings of the 3rd International Workshop on Socially-Aware Multimedia. 2014.

[10] Vijay, T., Chawla, A., Dhanka, B., Karmakar, P.: Sentiment analysis on covid-19Twitter data. In: 2020 5th IEEE International Conference on Recent Advances andInnovations in Engineering (ICRAIE), pp. 1–7. IEEE (2020)

[11] Elbagir, S., Yang, J.: Twitter sentiment analysis using natural language toolkitand VADER sentiment. In: Proceedings of the International Multiconference ofEngineers and Computer Scientists, vol. 122, p. 16 (2019)

[12] Hutto, C., Gilbert, E.: Vader: a parsimonious rule-based model for sentiment anal-ysis of social media text. In: Proceedings of the International AAAI Conference onWeb and Social Media, pp. 216–225 (2014)

[13] Joyce, B., Deng, J.: Sentiment analysis of tweets for the 2016 US presidential elec-tion. In: 2017 IEEE MIT Undergraduate Research Technology Conference (URTC),pp. 1–4. IEEE (2017)

[14] Han, Jun, and Claudio Moraga. "The influence of the sigmoid function parameters on the speed of ackpropagation learning." In International workshop on artificial neural networks, pp. 195-201. Springer, Berlin, Heidelberg, 1995

[15] Xia, E., Yue, H., Liu, H.: Tweet sentiment analysis of the 2020 US presidentialelection. In: Companion Proceedings of the Web Conference 2021, pp. 367–371(2021)

[16] Kaur, P., Edalati, M.: Sentiment analysis on electricity Twitter posts. arXivpreprint arXiv:2206.05042 (2022)

[17] E. Raisi, B. Huang, Weakly supervised cyberbullying detection with participant-vocabulary consistency. Social Network Analysis and Mining. 8 (2018), doi:10.1007/s13278-018-0517-y.

[18] Y. N. Silva, D. L. Hall, C. Rich, BullyBlocker: toward an interdisciplinary approach to identify cyberbullying. Social Network Analysis and Mining. 8 (2018), doi:10.1007/s13278-018-0496-z.

[19] T. Bin Abdur Rakib, L. K. Soon, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Springer Verlag, 2018), vol. 10751 LNAI, pp. 180–189.

[20] Homa Hosseinmardi, Sabrina Arredondo Mattson, Rahat Ibn Rafiq, Richard Han, Qin Lv, Shivakant Mishra. (2015). Detection of Cyberbullying Incidents on the Instagram Social Network."

[21] Dadvar, Maral Eckert, Kai. (2018). Cyberbullying Detection in Social Networks Using Deep Learning Based Models; A Reproducibility Study. 10.13140/RG.2.2.16187.87846.

[22] Nandhini, B. Sri, and J. I. Sheeba." Cyberbullying detection and classification using information retrieval algorithm." Proceedings of the 2015 International Conference on

[23] Hasib, Khan Md, Md Ahsan Habib, Nurul Akter Towhid, and Md Imran HossainShowrov. "A Novel deep Learning based Sentiment Analysis of Twitter Data for US Airline Service." In 2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD), pp. 450-455. IEEE, 2021.

[24] Neogi, A.S., Garg, K.A., Mishra, R.K., Dwivedi, Y.K.: Sentiment analysis andclassification of Indian farmers' protest using Twitter data. Int. J. Inform. Manage.Data Insights 1(2), 100019 (2021)

[25] Prabhat, A., Khullar, V.: Sentiment classification on big data using naive Bayesand logistic regression. In: 2017 International Conference on Computer Communi-cation and Informatics (ICCCI), pp. 1–5 (2017). https://doi.org/10.1109/ICCCI.2017.8117734

[26] Sefara, T.J., Mokgonyane, T.B.: Emotional speaker recognition based on machineand deep learning. In: 2020 2nd International Multidisciplinary Information Tech-nology and Engineering Conference (IMITEC), pp. 1–8. IEEE (2020)

[27] Medhat, W., Hassan, A., Korashy, H.: Sentiment analysis algorithms and applica-tions: a survey. Ain Shams Eng. J. 5(4), 1093–1113 (2014)

[28] Marivate, V., Sefara, T.: Improving short text classification through global aug-mentation methods. In: Holzinger, A., Kieseberg, P., Tjoa, A.M., Weippl, E. (eds.)CD-MAKE 2020. LNCS, vol. 12279, pp. 385–399. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-57321-8_21

[29] oyce, B., Deng, J.: Sentiment analysis of tweets for the 2016 US presidential elec-tion. In: 2017 IEEE MIT Undergraduate Research Technology Conference (URTC),pp. 1–4. IEEE (2017)

[30] Ramadhan, W., Astri Novianty, S., Casi Setianingsih, S.: Sentiment analysis usingmultinomial logistic regression. In: 2017 International Conference on Control, Elec-tronics, Renewable Energy and Communications (ICCREC), pp. 46–49 (2017).DOI: https://doi.org/10.1109/ICCEREC.2017.8226700

SENTIMENT ANALYSIS AND OFFENSIVE DETECTION

ORIGINALITY REPORT

13%	10%	3%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	6%
2	Tshephisho Joseph Sefara, Mapitsi Roseline Rangata. "Chapter 37 Domain-Specific Sentiment Analysis ofTweets Using Machine Learning Methods", Springer Science and Business Media LLC, 2024 Publication	2%
3	Submitted to Daffodil International University Student Paper	2%
4	dokumen.pub Internet Source	<1%
5	Asad Khattak, Anam Habib, Muhammad Zubair Asghar, Fazli Subhan, Imran Razzak, Ammara Habib. "Applying deep neural networks for user intention identification", Soft Computing, 2020 Publication	<1%
6	spectrum.library.concordia.ca Internet Source	<1%