Hate Speech, Offensive Language Detection and Blocking on Social Media Platform using Feature Engineering Techniques and Machine Learning Algorithms a Comparative Study

By

Mwayi Malemia
21251377011

Guide

Dr. Glorindall

Project Report

Submitted

In partial fulfillment of the requirements for the

Masters of Science in Computer Science
April, 2023

DMI
ST. EUGENE
Fully Alive, Fully Human

DMI - ST. EUGINE UNIVERSITY
ZAMBIA CAMPUS

ABSTRACT

The increasing use of social media and information sharing has given major benefits to humanity. However, this has also given rise to a variety of challenges including the spreading and sharing of hate speech messages. Thus, to solve this emerging issue in social media sites, recent studies employed a variety of feature engineering techniques and machine learning algorithms to automatically detect the hate speech messages on different datasets. However, to the best of my knowledge, not much research has been done to compare the variety of machine learning algorithms to evaluate which machine learning algorithm outshine on a standard publicly available dataset. Hence, the aim of this paper is to compare the performance of machine learning algorithms to appraise their performance on a publicly available dataset having three distinct classes. The study has proved that the bigram features when used with the support vector machine algorithm best performed with 79% off overall accuracy. My study holds practical implication and can be used as a baseline study in the area of detecting automatic hate speech messages.

TABLE OF CONTENTS

CHAPTER ONE
INTRODUCTION

Social networking sites are the most efficient way to meet new people. The assortment in content on social media has made people became creative and open minded. Social media has thus developed into really powerful medium to share ideas and opinions. However, due to the rapid growth and popularity of social networking sites, many users have discovered an illegal and immoral way to use them. The most commonly encountered and most dangerous misuses of online social media are the expression of hate and harassment.

Hate Speech in relation to social media, is a kind of writing that disparages and is likely to cause harm or danger to the victim. It is a bias-motivated, hostile, malicious speech aimed at a person or a group of people because of some of their actual or perceived innate characteristics [12]. It is a kind of speech that demonstrates a clear intention to be hurtful, to incite harm, or to promote hatred. The environment of social media provides a particularly fertile ground for creation, sharing and exchange of hate messages against a perceived enemy group [5].

However, identifying and removing hate speech content has proved to be labor-intensive and time consuming. Owing to these worries and prevalent hate speech content on the internet, there is a strong motivation to implement an automated hate speech detection system. The automatic detection of hate speech has proved to be a challenging task because of disagreements on different hate speech definitions as perceived by many. Detection of hate speech and offensive language has been considered as an emerging application in numerous research problems associated with the domain of Natural Language Processing [13]

➢ *Background of the study*

The abuse on social media has shown to be more and more of significant in the last decade hence, the process of noticing or eliminating such content manually from the web is a tedious task. So, there is a need of developing an automated model that is able to notice such toxic content on the web.

Regardless of the extensive amount of work that researchers have so far done, it remains problematic to make comparisons on the performance of these approaches to categorize hate speech content that is flooded on the social media. To my knowledge based on literature that I have read so far, the prevailing studies lack the comparative analysis of dissimilar feature engineering techniques and ML algorithms.

➢ *Statement of the problem*

Identifying and removing hate speech content using manual process has proved to be labor-intensive and also time consuming. Manual annotation and removal of the hate speech isn't possible because of the huge amount of data processed every second. For example, as of 2020, there are more than 6000 tweets sent every second [17].

➢ *Objectives of the study*

To manage the enormous volumes of data existing in this virtual sphere, there is an urgent requirement for intelligent systems that are capable of automatically able to flag and classify the content using numerous machine learning models and feature engineering techniques. In this paper, I propose an approach to devise machine learning models which will be combined together with feature engineering techniques and later evaluate their performance based on a publicly available dataset having three distinct classes for classification. Machine learning has lately been used in wide variety of fields like intelligent healthcare, smart homes, cybersecurity and many more [18].

• *Main Objective*

This paper will discuss the ways in which machine learning and feature engineering techniques are used to control hate speech and abusive language on social media with a given dataset.

• *Specific Objective*

My study is quite significant as it donates to resolving the problem at hand, by relating three feature engineering and eight Machine Learning classifiers on standard hate speech datasets having three distinct classifiers. The study holds applied reputation and therefore, serves as a base line for new researchers in the domain of automatic hate speech detection on social media platform.

➢ *Research Question*

Which combination of three feature engineering and eight Machine Learning classifiers out performs on a standard hate speech dataset?

➢ *Significance of the Study*

My study holds a practical implication and can be used as a baseline study in the area of detecting automatic hate speech messages.

- *Scope of the Study*

The study is focusing on three branches of Artificial Intelligence (AI), they include; (a) Machine Learning, (b) Deep Learning and (c) Natural Language Processing. Below, I have explained the basics of each topic and how they relate to my research.

- Artificial intelligence “(AI) is an area of computer science that emphasizes the creation of intelligent machines that work and reacts like humans”. (Andrew Ng, 2015)
- Machine learning “(ML) is the science of getting computers to learn and act like human do, and improve their learning over time in autonomous fashion, by feeding them data and information in the form of observations and real-word interactions”. (Arthur Samuel, 2016).
- Natural Language Processing is the sub-field that takes the inspiration from the areas of Artificial Intelligence and Linguistics. It enables the computers/machines to process and analyze the large amount of human language data such as speech or text.

- *Definition of Unfamiliar Terms.*

Table 1 Text Cataloging

S. No.	Concept	Abbreviation	Description
1	Feature Extraction	FE	It is mapping from text data to real valued vectors
2	Bigram	-	It's a feature engineering technique which represents two adjacent words in a single numeric feature while creating master feature vectors for words.
3	Term Frequency - Inverse Document Frequency	TFIDF	It's a feature representation technique that represents “word importance” is to a document in the document set. It works in a combination of the frequency of word appearance in a document with no. of documents containing that word.
4	Word2vec		It is a technique used to learn vector representation of words, which can further be used to train machine learning models
5	Doc2vec		It is an unsupervised technique to learn document representations in fixed-length vectors. It is the same as word2vec, but the only difference is that it is unique among all documents.
6	Machine Learning Classifiers	ML Classifiers	These are applied to numeric features vector to build the predictive model which can be used for prediction class labels.
7	Naïve Bayes	NB	It's a probabilistic based classification algorithm, which uses the “Bayes theorem” to predict the class. It works on conditional independence among features.
8	Random Forest	RF	It's a type of ensemble classifier consisting of many decision trees. It classifies an instance based on voting decision of each decision trees class predictions.
9	Support Vector Machines	SVM	It's a supervised classification algorithm which constructs an optimal hyperplane by learning from training data which separates the categories while classifying new data.
10	K Nearest Neighbor	KNN	It's a simple text classification algorithm, which categorize the new data using some similarity measure by comparing it with all available data.
11	Decision Tree	DT	It is a supervised algorithm. It generates the classification rules in the tree-shaped form, where each internal node denotes attribute conditions, each branch denotes conditions for outcome and leaf node represents the class label.
12	Adaptive Boosting	AdaBoost	It is one of the best-boosting algorithms, which strengthens the weak learning algorithms.
13	Multilayer Perceptron	MLP	It is a feedforward artificial neural network. It produces a set of outputs using a set of inputs
14	Logistic Regression	LR	It is a predictive analysis. It uses a sigmoid function to explain the relationship between one independent variable and one or more independent variables

CHAPTER TWO
LITERATURE REVIEW

➢ *Introduction*

Research has proved that a lot of study has been done from across the world on hate speech and abusive language recognition written in diverse languages such as English, Dutch, French, Hindi, Greek. Several studies have so far been concluded in the field of the detection of hate speech and abusive language using machine learning models, deep learning architectures, language models, etc. To one side, while recognizing the variety in models and architectures, different works have different data which are annotated for different aspects or labels. Similarly, the dataset might be in different languages. Each language has their own lexical, morphological, and syntactic structures.

➢ *Main Literature Review*

Gaydhani et al., 2018 [6] They proposed a solution to the detection of hate speech and offensive language on Twitter through machine learning using n-gram features weighted with TFIDF values. They performed comparative analysis of Logistic Regression, Naive Bayes and Support Vector Machines on various sets of feature values and model hyper parameters. The results showed that Logistic Regression performs better with the optimal ngram range 1 to 3 for the L2 normalization of TFIDF. Upon evaluating the model on test data, we achieved 95.6% accuracy. It was seen that 4.8% of the offensive tweets were misclassified as hateful. This problem can be solved by obtaining more examples of offensive language which does not contain hateful words. The results can be further improved by increasing the recall for the offensive class and precision for the hateful class. Also, it was seen that the model does not account for negative words present in a sentence. Improvements can be done in this area by incorporating linguistic features.

Parihar et al., 2021 [18] Hate speech detection is a very difficult task and continues to be a societal problem. There is a very fine line between what is a hate speech and what is not. For example, a satire might also be considered as a possible threat but it is not actually a hate speech. The annotation and collection of data for building a model for hate speech detection is thus a very troublesome task. As discussed, this problem can be solved by narrowing down the criteria for annotations. Similarly, there is a need to focus research on code-mixed languages and regional languages as well. Language models and deep learning models have shown promising results in hate speech classifications. For tackling with unbalanced data, the up sampling or down sampling techniques based on language models should be researched upon. The challenges discussed above must be tackled with more research in the domain so that the internet becomes more inclusive, welcoming and free from hate.

Mahibha et al, [13] They concluded that, from the output obtained from the different models it could be inferred that deep learning models outperform the machine learning models considering the offensive language classification problem for the data set provided by HASOC@FIRE-2021 for Task 1 associated with code mixed Tamil. Among the deep learning model transformer-based models has done the more accurate predictions compared to recurrent models, hence more scope for transformer-based models could be identified for research based on Dravidian languages and in specific Hate and Offensive language-based researches.

Zeerak Waseem et al. [20] classify the hate speech on twitter. In their research, they employed character Ngrams feature engineering techniques to generate the numeric vectors. The authors fed the generated numeric vector to the LR classifier and obtained overall 73% F-score. While, Chikashi Nobata et al. [6] used the ML -based approach to detect the abusive language in online user content. In their research authors employed character Ngrams feature representation technique to represent the features. The authors fed the features to the SVM classifier. The results showed that the classifier obtained overall 77% F-score. Shervin Malmasi et al [14] used an ML -based approach to classify hate speech in social media. In their research, the authors employed 4grams with character grams feature engineering techniques to generate numeric features. The authors fed the generated numeric features to the SVM classifier. The authors reported maximum of 78% accuracy.

Al-Hassan and Al-Dossari, 2019 [1] Arab regions and worldwide are now more aware of the problem of spreading hate through the social networks. Many countries are working hard in regulating and countering such speech. This attention raised the need for automating the detection of hate speech. In this paper we analyzed the concept of hate speech and specifically “cyber hate” which is conducted in the means of social media and the internet sphere. Moreover, they differentiated between the different anti-social behaviors which include (Cyberbullying, Abusive and offensive language, Radicalization and hate speech). After that they presented a comprehensive study on how text mining can be used in social networks. we investigated some challenges which can be a guide for the implementation of Arabic hate speech detection model. In addition, these recommendations will help in drawing a road map and a blueprint for the future model. The future work will include incorporating the latest deep learning architectures to build a model that is capable to detect and classify Arabic hate speech in twitter into distinct classes. A data set will be collected from twitter, and for intensifying the training of our neural network they will including data from additional platform “e.g. Facebook” as it is the most used platform in the Arab region.

Mesa-Jimenez et al., 2022 [15]. In their findings they present a two-stage text classification study in the field of BMS. The results of the first stage show that XGBoost performs better than the other four, but the others make good candidates for this stage too, except maybe for multinomial Naive Bayes, which shows slightly worse results. The outperformer in the second stage classification, the multi label problem, is logistic regression. The top performers that follows are followed by XGBoost algorithm and, again, the Naive Bayes method performs the worst of the five. The accuracy per tag type shows that certain algorithms may be better in predicting certain tags than others. In the current paper, we have considered XGBoost and logistic regression to design the system, but the aim for further work will be a combination of methods for the second stage, using each method for doing only the classifications they are the best at, to improve the general accuracy of the whole implementation. Sub-dividing the problem into several problems improves its accuracy for the whole system as expected. In terms of the model's deployment, the assessment of errors is very important. The main problem for this system's implementation is to locate false positive elements. The false positives are the incorrectly tagged elements that passed to the building analytics software. These elements may be difficult to detect, especially for buildings with a big number of points. Increasing the confidence boundary to a higher level may help to solve this problem and reduce false positives to a minimum. This also may reduce the number of true positives, increasing the amount of manual work. The findings of this work open a new field of application for text classification methodologies, aiming to a scientific audience, which may explore the methodologies of this paper further to generalize this field of application for text processing and categorization, or to industrial professionals who seek to implement this system to reduce operational tagging times from several days to a couple of minutes. Our research provides a novel multi-stage machine learning solution for the real-world BMS problem, which can be applied in several systems, or even re-trained with new standards that could appear in the future.

CHAPTER THREE
RESEARCH DESIGN

➢ *Introduction*

This section represents a general methodology for building hate speech detection model. The process starts with dataset collection and goes through the process of annotation or labelling of the data, extraction of features, use of learning algorithms and evaluation of performance.

➢ *Research Design*

The section explains the Methodology that has been adopted in order to categorize tweets into three different distinct classes namely, "hate speech, offensive but not hate speech, and neither hate speech nor offensive speech". Fig. 1 below illustrates a comprehensive research methodology that has been implemented in this research. As presented by the figure below, the methodology employed has six key steps that are going to be used before the results can be concluded. The steps include; (1) data collection, (2) data preprocessing, (3) feature engineering, (4) data splitting, (5) classification model construction, (6) and classification model evaluation. Each step is explained in detail below.

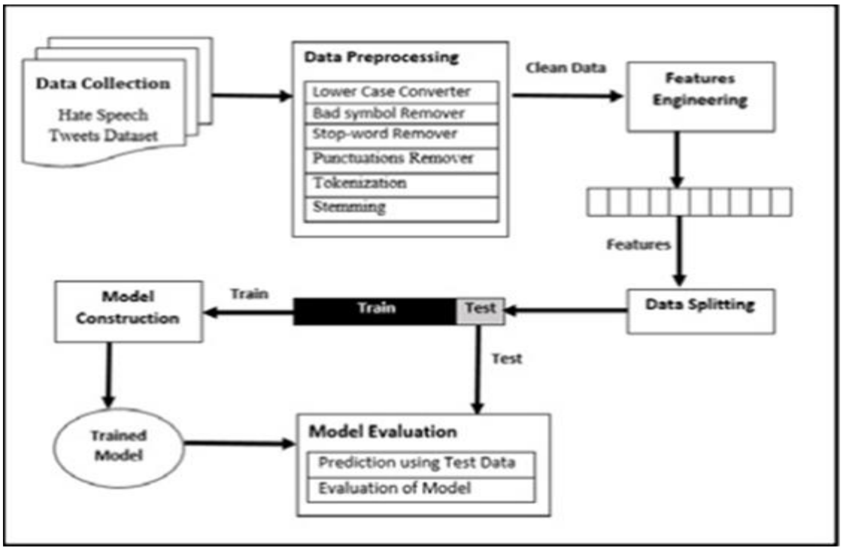


Fig 1 Research Methodology, step by step process

- *Data Collection*

This study, for the purpose collecting research data, I have used a publicly available open source CrowdFlower dataset. CrowdFlower provided this dataset as an open source, they compiled and labelled datasets making it very user friendly. The tweets are labeled into three distinct classes, namely, hate speech, not offensive, and offensive but not hate speech. This dataset has 14509 number of tweets. Of these, 16% of tweets belong to class hate speech. In addition, 50% of tweets belong to not offensive class and the remaining 33% tweets are offensive but not hate speech class. The details of this distribution are also shown in

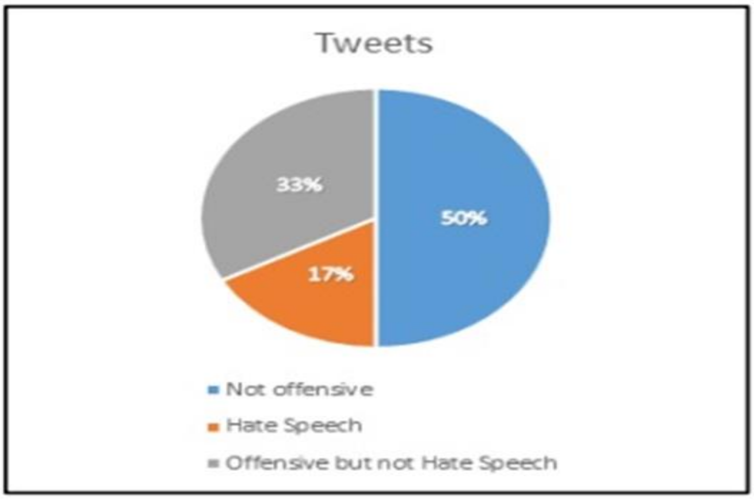


Fig 2 Crowd Flower Dataset Classification

- *Text Preprocessing*

Different preprocessing-techniques have been used in order to sieve noisy and non-informative features from the tweets. The preprocessing technique involves transforming tweets into lower case. The technique correspondingly, is applied to remove all the URLs, usernames, white spaces, hashtags, punctuations and stop-words using pattern matching techniques from the collected tweets. Subsequent to preprocessing the tweets also undergo Tokenization and stemming. The tokenization, is the process that is applied to convert each single tweet into tokens or words, then the porter stemmer converts words to their root forms, such as offended to offend using porter stemmer.

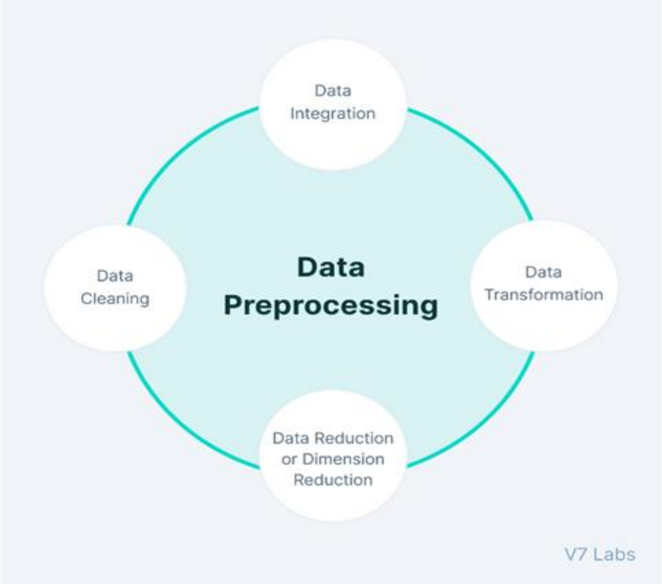


Fig 3 Data Preprocessing Cycle

- *Feature Engineering*

Basically, ML algorithms do not recognize the classification rules from the raw text. Hence the need for numerical features to understand classification rules. This is why feature engineering is highly considered as one among the top key steps in text classification. At this level it where key features are extracted from raw text and later representing the extracted features in numerical form. This study has therefore, used three unalike feature engineering techniques. They include; n-gram with TFIDF, Word2vec and Doc2vec.

- *Data Splitting*

The table below illustrates the class-wise distribution and also results after splitting of the overall dataset. The table is showing the number of instances that has been used in Training set and also the number of instances that has been used in Test set. In the study to split the preprocessed data we have used 80-20 ratio, basically what it means is that 80% of the instances has been used for Training Data while 20% has been dedicated for Test Data. The whole idea is to ensure that classification models are trained to learn classification rules.

Table 2 Details of Data Split

S no.	Category	Total Insurances	Training Numbers	Testing Numbers
0	Hateful Speech	2397	1908	489
1	Not offensive Speech	7275	5814	1459
2	Offensive but not Hateful Speech	4837	3884	954
	Total	**14509**	**1607**	**2902**

- *Machine Learning Models*

It is significantly commended to apply several unlike classifiers on a master feature vector to detect which one reaches to the better outcomes. This is because research has proved that there is no any single classifier which best performs on all kinds of dataset. This is the reason why eight different classifiers have been used for the purpose of this study. These include; NB [21], SVM [9], KNN [20], DT [2], RF , AdaBoost, MLP and LR [10].

- *Classifier Evaluation*

At this stage, it is where classes of unlabeled text are predicted by the constructed classifier. The Test Set process follows that the text is labeled into three distinct classes, namely; (0) hate speech, (1) offensive but not hate speech, (3) neither hate speech nor offensive speech. The matrixes of True Negative (TN), False Positives (FP), False Negatives (FN) and True Positives (TP) are calculated in order to evaluate classifier performance..

- *Population of the Study*

The study is targeting social media users from “Crowd Flower” an open source dataset with a population of 14509 records.

- *Sampling Procedure*

For the purpose of this study a **Probability Sampling** technique has been adopted in which samples from a larger population based on the theory of probability has equal chances. This sampling method considers every member of the population and forms samples based on a fixed process.

- *Sample Size*

The CrowdFlower dataset has a population of 14,509 number of tweets. This is an open source data which has been provided freely by CrowdFlower for the purposes of research and other studies.

- *Sampling Area*

Social media users via world wide web drawn from “Crowd Flower” dataset

- *Sources of Data Collection*

The dataset used in this study has been provided by “Crowd Flower” as an open source.

- *Methods of Data Collection*

“Crowd Flower” is an open source dataset which is publicly available to any interested person who has interest to use the data for research purposes.

- *Tools for Data Collection*

Since the dataset has been obtained from open source “Crowd Flower” it is apparently difficult to establish the tools that were used when collecting the data.

- *Tools for Data Analysis*

In this study Feature Engineering (FE) techniques together with 8 Machine Learning (ML) algorithms has been used to analyze data for research findings.

- *Limitations of the Study*

My study has found that most the limitations relating to the research are very much centered of feature extraction while classifying labels. Labels has a fixed maximum length which is 11words, and in many cases the labels may contain text full of errors, therefore it has proved to be extremely difficult and very challenging while extracting information in order to perform text classification into the 3 distinct classes. The other limitations is basically on inconsistencies that are found in the training datasets after performing data splitting process, hence manually tagged data are used in the system for training the models. The interpretation is that some engineers while running the models may use slightly different tags sometimes or simply that the information contained within the label itself is just incomplete and only compensated by personal experience, which limits the system results.

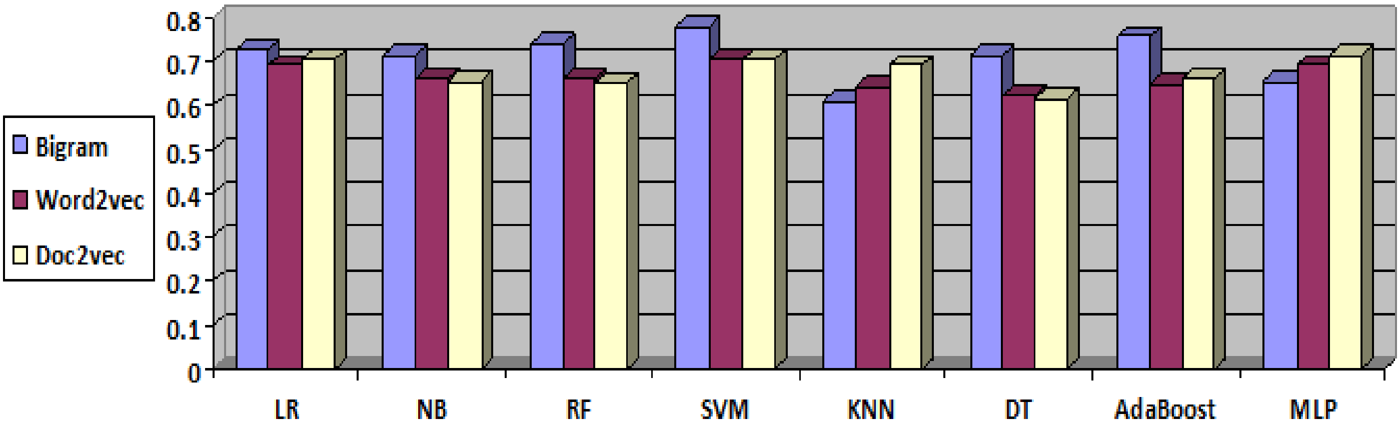
ISSN No:-2456-2165

CHAPTER FOUR
RESULTS

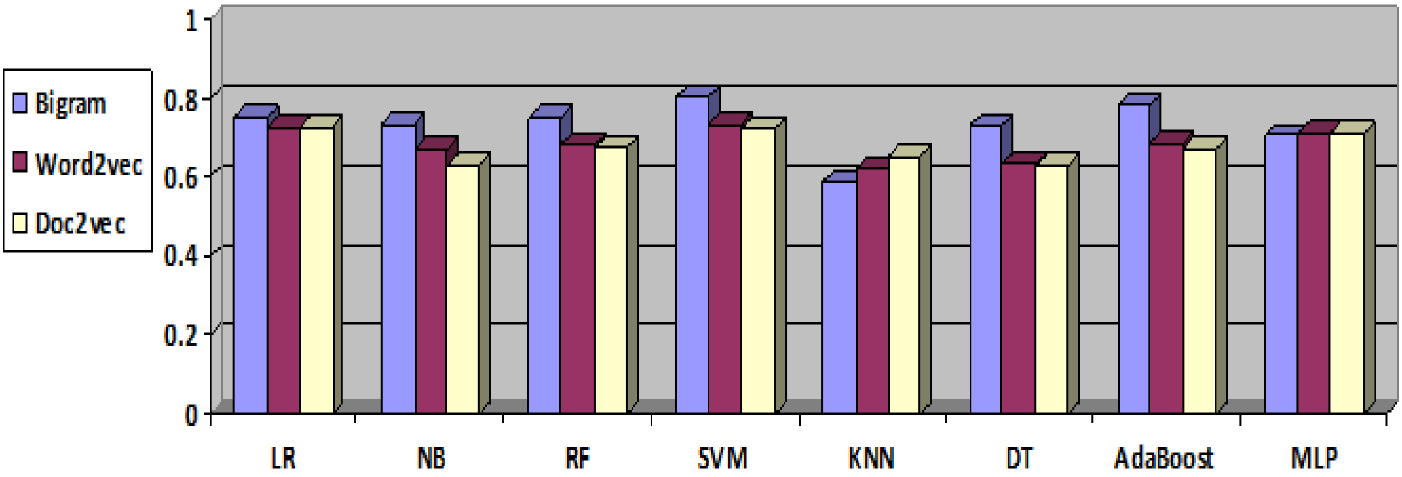
➢ *Introduction*

In this chapter I will talk about experimental settings that have been used in this study, as mentioned earlier I have used three types of feature engineering namely n-gram (bigram) with TFIDF, Word2vec and Doc2vec. Therefore, a total of three feature engineering representations have been computed, and also, eight ML algorithms were used to create three master feature vectors. As a result, overall 24 analyses (3 master feature vectors x 8 ML algorithms) were evaluated in order to check the effectiveness of classification models.

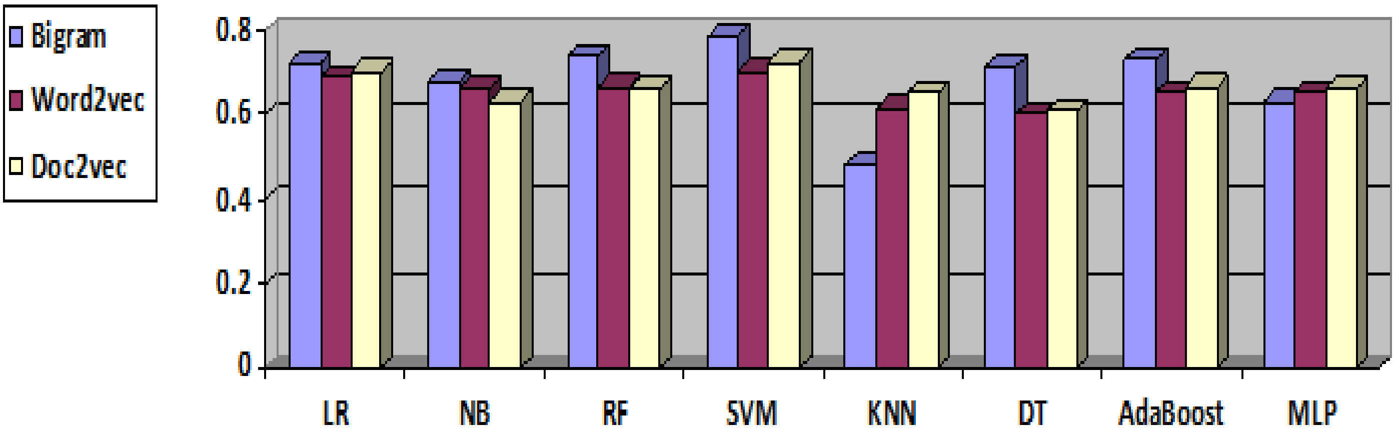
➢ *Graphs*



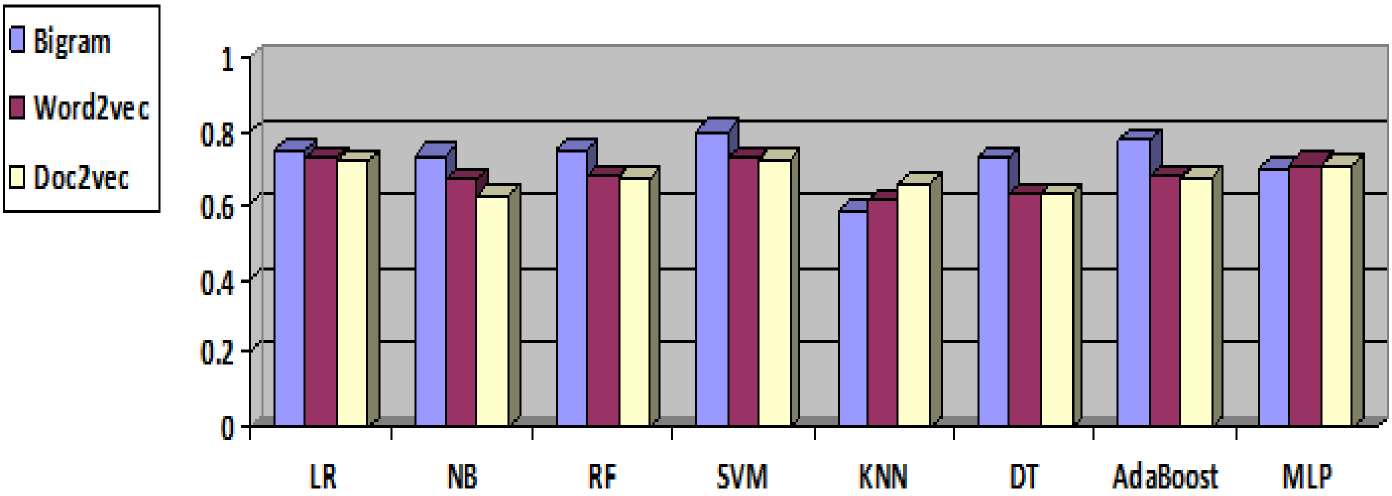
Graph A. Precision Analysis



Graph B. Recall Analysis



Graph C. F-Measure Analysis



Graph D. Accuracy Analysis

- *Interpretation*

This segment will explain the total results of 24 studies that have been conducted in the in this research. Graph A to graph D demonstrates (A) the precision, (B) recall, (C) F-measure and (D) accuracy of all 24 studies, respectively. The performance and classification techniques for each of the different feature representation are displayed graphically. Machine Learning (ML) algorithms with MLP and also KNN registered the **lowest results**. i.e. Precision analysis recorded 0.56, Recall analysis recorded 0.58, F-measure analysis recorded 0.48, while Accuracy analysis recorded 58%. Machine Learning (ML) algorithms with SVM while using TFIDF feature representation with bigram registered the **highest results.** Precision analysis recorded 0.78, Recall analysis recorded 0.80, F-measure analysis recorded 0.78, while Accuracy analysis recorded 80%. Feature engineering representation; Best performance has been registered with Bigram feature as compared to Word2vec and Doc2vec. However further analysis has revealed a marginal difference recorded from results obtained in bigram and Doc2vec. SVM classifier registered best performance in the **text classification** models. SVM results out classed all the eight classifiers. Never the less AdaBoost and RF classifiers results were lesser than SVM results and were better than LR, DT, NB, KNN, and MLP results.

CHAPTER FIVE
DISCUSSIONS OF FINDINGS

As a background to my study I have weighed three different feature engineering techniques over eight Machine Learning (ML) classifiers. Hence at the end obtaining results from an output of twenty-four analysis namely; (a) the precision analysis, (b) recall analysis, (c) F-measure analysis (d) accuracy analysis. The findings have revealed SVM algorithm as the best model while, working with combination of bigram feature engineering with TFIDF FE techniques.

- *Feature Engineering*

This study has three distinct feature extraction techniques which have been deployed and evaluated in their performance over a standard dataset. The three feature techniques include; Bigram with TFIDF, word2vec and doc2vec. The findings have revealed **Bigram with TFIDF** as the best performing model while Word2vec and Doc2vec has inversely performed lower. Bigram with TFIDF feature technique maintains a sequence of words unlike Word2vec and Doc2vec, this is probably the reason why Bigram with TFIDF performed better than the rest. Numerous research studies have showed that the TFIDF representation technique is better than the binary and term frequency representation Mujtaba et al., 2018 [16]. The likely cause for the lower performance of Word2vec is because it is unable to handle OOV (out of vocabulary) words specially in the domain of Twitter data.

- *Machine Learning*

Previous research has proved that "no single Machine Learning (ML) algorithm has performed better results on all kinds of dataset. It is against this background that several different ML algorithms have to be deployed in order to determine the best performer on a given dataset. SVM uses threshold functions to perform data separation and not the number of feature based on margin, this study has thus revealed SVM and AdaBoost as best classifiers based on this reason. This shows that SVM is independent upon the presence of the number of features in the data Hornik et al., 2013 [8]. The Kernel functions in SVM gives it ability to perform much better on non-linear data apart from working with linear data. The adaptive algorithms in AdaBoost enables the model to learn the classification rules, with much attention focusing on decreasing training error. This is the reason why AdaBoost has better performance as compared with the rest of the ML algorithms. The study has also revealed that SVM and AdaBoost classifiers has better results and on the second tier there is RF and LR who also have performed higher than results of NB, DT, KNN, and MLP which are placed on third tier. The absence of informative features in RF which result to incorrect predictions could be the reason for its low performance. It is possible that the performance of LR might be lower because its decision surface is linear in nature and cannot handle nonlinear data adequately Eftekhar et al., 2005 [5]. The reason behind the poor performance of the MLP classifier is due to not having enough training data that's why it is considered as complex "black box" Singh and Shahid Husain, 2014 [19]. The KNN had the worst performance due to laziness of the learning algorithm and it does not work adequately for noisy data Bhatia and Author, 2010 [3]. Therefore, according to this study the KNN has proved to be not suitable for detecting hate speech tweets.

- *Areas for Further Research*

My work has two distinct boundaries. First, the proposed ML model is inefficient in terms of real-time predictions accuracy for the data. Finally, it only classifies the hate speech message in three different classes and is not capable enough to identify the severity of the message. Henceforth, in the future, the objective is to improve the proposed ML model which can be used to predict the severity of the hate speech message as well.

CHAPTER SIX
CONCLUSION

The study has concluded that Bigram combined with TFIDF performed much better than Word2vec and Doc2vec feature engineering techniques. Furthermore, SVM and RF algorithms in Machine Learning (ML) algorithms have proved better results when compared with LR, NB, KNN, DT, AdaBoost, and MLP. The lowest performance was observed in KNN. The outcomes from this research study hold practical importance because this will be used as a baseline study to compare upcoming researches within different automatic text classification methods for automatic hate speech detection. Furthermore, this study also holds a scientific value because this study presents experimental results in form of more than one scientific measure used for automatic text classification.

REFERENCES

[1]. Areej Al-Hassan and Hmood Al-Dossari. 2019. DETECTION OF HATE SPEECH IN SOCIAL NETWORKS: A SURVEY ON MULTILINGUAL CORPUS. In *Computer Science & Information Technology (CS & IT)*, AIRCC Publishing Corporation, 83–100. DOI:https://doi.org/10.5121/csit.2019.90208

[2]. Asma Ben Abacha, Md. Faisal Mahbub Chowdhury, Aikaterini Karanasiou, Yassine Mrabet, Alberto Lavelli, and Pierre Zweigenbaum. 2015. Text mining for pharmacovigilance: Using machine learning for drug name recognition and drug–drug interaction extraction and classification. *Journal of Biomedical Informatics* 58, (December 2015), 122–132. DOI:https://doi.org/10.1016/j.jbi.2015.09.015

[3]. Nitin Bhatia and Corres Author. 2010. Survey of Nearest Neighbor Techniques. 8, 2 (2010).

[4]. Ying Chen, Yilu Zhou, Sencun Zhu, and Heng Xu. 2012. Detecting Offensive Language in Social Media to Protect Adolescent Online Safety. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, IEEE, Amsterdam, Netherlands, 71–80. DOI:https://doi.org/10.1109/SocialCom-PASSAT.2012.55

[5]. Behzad Eftekhar, Kazem Mohammad, Hassan Eftekhar Ardebili, Mohammad Ghodsi, and Ebrahim Ketabchi. 2005. Comparison of artificial neural network and logistic regression models for prediction of mortality in head trauma based on initial clinical data. *BMC Med Inform Decis Mak* 5, 1 (December 2005), 3. DOI:https://doi.org/10.1186/1472-6947-5-3

[6]. Aditya Gaydhani, Vikrant Doma, Shrikant Kendre, and Laxmi Bhagwat. 2018. Detecting Hate Speech and Offensive Language on Twitter using Machine Learning: An N-gram and TFIDF based Approach. Retrieved April 10, 2023 from http://arxiv.org/abs/1809.08651

[7]. Njagi Dennis Gitari, Zuping Zhang, Hanyurwimfura Damien, and Jun Long. 2015. A Lexicon-based Approach for Hate Speech Detection. *IJMUE* 10, 4 (April 2015), 215–230. DOI:https://doi.org/10.14257/ijmue.2015.10.4.21

[8]. Kurt Hornik, Patrick Mair, Johannes Rauch, Wilhelm Geiger, Christian Buchta, and Ingo Feinerer. 2013. The **textcat** Package for n -Gram Based Text Categorization in *R. J. Stat. Soft.* 52, 6 (2013). DOI:https://doi.org/10.18637/jss.v052.i06

[9]. Thorsten Joachims. 1998. Text categorization with Support Vector Machines: Learning with many relevant features. In *Machine Learning: ECML-98*, Claire Nédellec and Céline Rouveirol (eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 137–142. DOI:https://doi.org/10.1007/BFb0026683

[10]. Hui Li and Zeming Li. 2022. Text Classification Based on Machine Learning and Natural Language Processing Algorithms. *Wireless Communications and Mobile Computing* 2022, (July 2022), 1–12. DOI:https://doi.org/10.1155/ 2022/ 3915491

[11]. Shuhua Liu and Thomas Forss. 2014. Combining N-gram based Similarity Analysis with Sentiment Analysis in Web Content Classification: In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval*, SCITEPRESS - Science and and Technology Publications, Rome, Italy, 530–537. DOI:https://doi.org/10.5220/ 0005170305300537

[12]. Sean MacAvaney, Hao-Ren Yao, Eugene Yang, Katina Russell, Nazli Goharian, and Ophir Frieder. 2019. Hate speech detection: Challenges and solutions. *PLoS ONE* 14, 8 (August 2019), e0221152. DOI:https://doi.org/10.1371/ journal.pone. 0221152

[13]. C Jerin Mahibha, Sampath Kayalvizhi, Durairaj Thenmozhi, and Sundar Arunima. Offensive Language Identification using Machine Learning and Deep Learning Techniques.

[14]. Shervin Malmasi and Marcos Zampieri. 2017. Detecting Hate Speech in Social Media. Retrieved April 11, 2023 from http://arxiv.org/abs/1712.06427

[15]. Jose Joaquin Mesa-Jiménez, Lee Stokes, QingPing Yang, and Valerie N. Livina. 2022. MACHINE LEARNING FOR TEXT CLASSIFICATION IN BUILDING MANAGEMENT SYSTEMS. *JOURNAL OF CIVIL ENGINEERING AND MANAGEMENT* 28, 5 (May 2022), 408–421. DOI:https://doi.org/10.3846/jcem.2022.16012

[16]. Ghulam Mujtaba, Liyana Shuib, Ram Gopal Raj, Retnagowri Rajandram, and Khairunisa Shaikh. 2018. Prediction of cause of death from forensic autopsy reports using text classification techniques: A comparative study. *Journal of Forensic and Legal Medicine* 57, (July 2018), 41–50. DOI:https://doi.org/10.1016/j.jflm.2017.07.001

[17]. Keisuke Nakagawa, Nuen Tsang Yang, Machelle Wilson, and Peter Yellowlees. 2022. Twitter Usage Among Physicians From 2016 to 2020: Algorithm Development and Longitudinal Analysis Study. *J Med Internet Res* 24, 9 (September 2022), e37752. DOI:https://doi.org/10.2196/37752

[18]. Anil Singh Parihar, Surendrabikram Thapa, and Sushruti Mishra. 2021. Hate Speech Detection Using Natural Language Processing: Applications and Challenges. In *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)*, IEEE, Tirunelveli, India, 1302–1308. DOI:https://doi.org/10.1109/ICOEI51242.2021.9452882

[19]. Pravesh Kumar Singh and Mohd Shahid Husain. 2014. Methodological Study Of Opinion Mining And Sentiment Analysis Techniques. *IJSC* 5, 1 (February 2014), 11–21. DOI:https://doi.org/10.5121/ijsc.2014.5102

[20]. Zeerak Waseem and Dirk Hovy. 2016. Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*, Association for Computational Linguistics, San Diego, California, 88–93. DOI:https://doi.org/10.18653/v1/N16-2013

[21]. Baoxun Xu, Xiufeng Guo, Yunming Ye, and Jiefeng Cheng. 2012. An Improved Random Forest Classifier for Text Categorization. *JCP* 7, 12 (December 2012), 2913–2920. DOI:https://doi.org/10.4304/jcp.7.12.2913-2920

[22]. View of Modeling the Detection of Textual Cyberbullying. Retrieved April 11, 2023 from https://ojs.aaai.org/index.php/ICWSM/article/view/14209/14058