A Voting enabled Predictive Approach for Hate Speech Detection

Parnani Panda
School of Computer Engineering
Kalinga Institute of Industrial Technology
Bhubaneswar, India
parnanipanda18@gmail.com

Sushruta Mishra
School of Computer Engineering
Kalinga Institute of Industrial Technology
Bhubaneswar, India
sushruta.mishrafcs@kiit.ac.in

Vandana Sharma
Computer Science Department
CHRIST (Deemed to be University)
Delhi NCR, India
vandana.juyal@gmail.com

Ahmed Alkhayyat
College of Technical Engineering
The Islamic University
Najaf, Iraq.
ahmedalkhayyat85@gmail.com

2023 6th International Conference on Contemporary Computing and Informatics (IC3I) | 979-8-3503-0448-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/IC3I59117.2023.10397912

***Abstract*- In today's digital environment, hate speech, which is defined as disparaging and discriminating communication based on personal characteristics, presents a big difficulty. Hate crimes and the rising amount of such content on social media platforms are two examples of how it is having an impact. Large volumes of textual data require manual analysis and categorization, which is tedious and subject to prejudice. Machine learning (ML) technologies have the ability to automate hate speech identification with increased objectivity and accuracy in order to overcome these constraints. This article intends to give a comparative analysis of various ML models for the identification of hate speech. The proliferation of such content online and its negative repercussions on people and society are explored, as is the necessity for automated hate speech recognition. This paper intends to support the creation of efficient hate speech detection systems by performing a comparative analysis of ML models. Random forest records the best performance with higher accuracy and low response delay period for hate speech detection. The results will help enhance automated text classification algorithms and, in the end, promote a safer and more welcoming online environment by illuminating the benefits and drawbacks of various approaches.**

Keywords - Hate speech detection, machine learning, logistic regression, Naive Bayes classifier, Random Forest classifier, CountVectorizer.

I. INTRODUCTION

Hate speech is described as any kind of communication which affects an individual or a group of individuals based on a traits such as color, race, ethnicity, gender, sexual orientation, nationality, religion, or other characteristics [1]. Hate crimes are increasingly involving social media. For example, video footage from the suspect in the New Zealand terror incident in 2019 was posted live on Facebook [2]. In this era of big data, manually analyzing and categorizing massive volumes of textual data is time-consuming and difficult [3]. Human variables such as fatigue and competence can also have an impact on the accuracy of manual text classification. Machine learning (ML) technologies will help to automate text classification procedures in order to produce better and less subjective results [4]. Most reports demonstrate the numerous ways that technology is producing the kind of progress that our world requires. They did better than ever before in the final three months of 2020 to detect hate speech, and bullying - 97% of hate speech removed from Facebook was detected by computerized systems before it was flagged by a human, which is an increase from 94% in the preceding quarter and 80.5% in 2019. This paper presents a comparative study of different models.

A. Motivation

Hate speech has been more prevalent in recent years, both in person and online. At the very least, it generates despair, a sense of neglect, aggravation, unwarranted rage, and hatred in the victim's thoughts, as well as onlookers who witness the act. In some cases, it has resulted in hate crimes, wherein the affected individuals acted on some of their negative emotions as a result of the increased emotional conflict sparked by such derogatory remarks, creating victims of spiraling events that began with just a single person's remark on a sensitive matter. Various groups have taken various actions, including laws, to combat the destructive effects of hate speech. The manual process of finding and eliminating hate speech content, on the other hand, takes time and effort. There is a lot of incentive for computerized hate speech identification since there is so much hate speech material on the internet.

II. LITERATURE REVIEW

Using machine learning and deep learning techniques, [5] explored the detection of hate speech in texts. The study aimed to search hateful and insulting words in social media messages on Twitter and Facebook. They used Deep Learning techniques like Logistics Regression and Naive Bayes classifiers, as well as Machine Learning techniques like Logistics Regression and Naïve Bayes classifiers. For their trials, they employed two approaches: the first used

979-8-3503-0448-0/22/$31.00 ©2023 IEEE

CountVectorizer and Tfidf to represent the input features, and the second used CountVectorizer and Tfidf to represent the output features. Logistic regression and Naïve-Bayes classifiers were taught using the features. Their second strategy employed three layers of ID Convolutional using CNN. Data pre-processing was not used in their trials. In any of the sub-tasks employed, their approaches' accuracy did not reach 70%. In [6] classified hate contents in social media with predictive learning models. Authors designed a metadata driven approach to develop a meta-attribute matrix. The features were built with emotions analogies, syntactic and objective variables with a bias for hate content. The best attributes were subjected to a hybrid classification model where it was validated in practical set up with a precise parameter thereby receiving 73% metric. In [7] employed a dictionary-based method to identify cyber hate on Twitter. Using an multi-gram attribute selection method, they created non-categorical features with a predefined dictionary of harsh terms. The authors used an ML classifier called SVM to feed the created numeric vector and got a maximum F-score of 67%. [8] used the supervised ML approach to categorise racist material. They employed a bigram feature extraction method to convert raw text into numeric vectors. The SVM classifier was used to execute the experiments. They were able to reach an accuracy of 87% in their results. To detect racism towards black people on Twitter, [9] utilised a machine learning-based technique. To produce numeric vectors, they used unigram in conjunction with a BOW-based approach. The authors used the Nave Bayes classifier to classify the numeric vector. Their findings were accurate to within 76% of the time. Hate speech on Twitter was classified by [10]. They used BOW characteristics in their investigation. The authors used the Nave Bayes classifier to classify the numeric vector. The results of their experiments demonstrated an accurate result of 73%.The ML-based technique was utilized in [11] to identify abusive language in the material on the internet. To depict the characteristics in their study, the scientists employed the character N-Grams feature extraction approach. These features were input into the SVM classifier by the authors. The classifier had an overall F-score of 77%, according to the data. In 2017, [12] conducted a survey on natural language processing to detect hate speech. The authors went into considerable length about investigations on various feature engineering methodologies for supervised categorization of hate speech texts. The biggest issue in the survey is that no experimental findings for the strategies were reported. [13] used social media comments or posts to classify sensitive themes. They employed unigram in conjunction with the TFIDF feature approach to create the numeric feature vectors. Four distinct classifiers were used to process the obtained features: Nave Bayes, rule-based, J48, and SVM. In their tests, the rule-based classifier outperformed the NB, J48, and SVM classifiers by 73 percent. [14] classified web content sites into two categories: hatred and violence. They made use of trigram features, which were showcased using TF IDF, in their research. The Naïve Bayes Classifier was used by the authors. The Naïve Bayes classifier achieved the best accuracy of 68%.

III. Proposed Framework

This section outlines the system for categorizing tweets into distinct categories that we used. The entire research approach has been shown in the image labeled as Figure 1. The study technique has been divided into six main parts, as indicated in this diagram: data collecting, data analysis, data preprocessing, feature engineering, data splitting, and model comparison. The next sections go over each of the steps in depth.

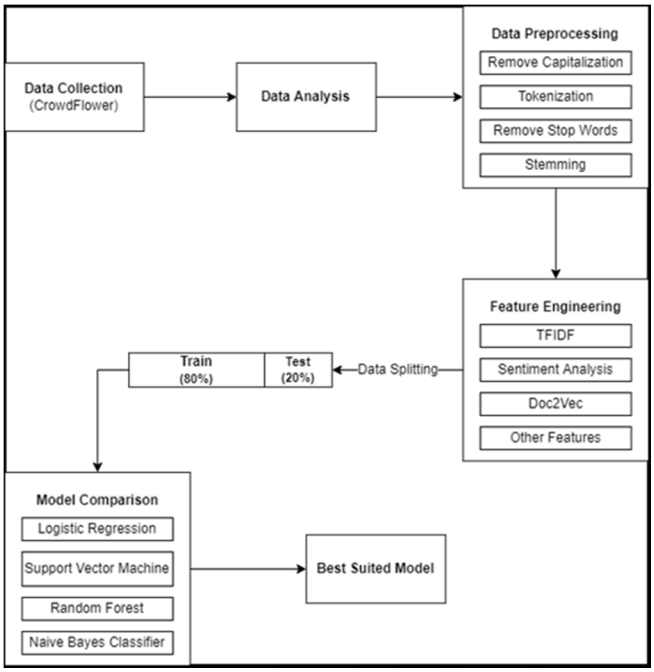

Figure 1. Proposed votinng based workflow model

A. Data Aggregation

We used CrowdFlower's publicly available dataset, which included manual classification of tweets into three different categories: hate speech, offensive language, and neither [15-16].

The histogram in Figure 2 shows that most of the tweets are considered to be offensive words by the CrowdFlower coders.

0: Hate Speech

1: Offensive Language

2: Neither

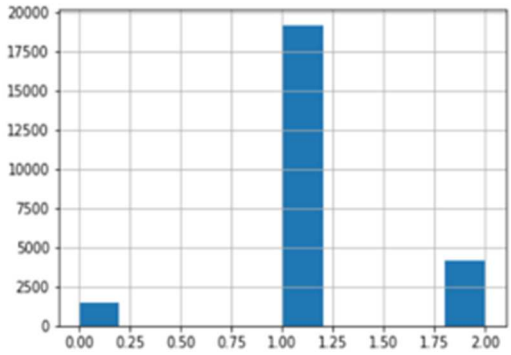

Figure 2. Offensive tweets as per CrowdFlower

B. *Data Analysis*

We added text length as a field and plotted it in graphs to conclude that text length distribution is the same among all three classes, but the number of tweets is turned higher towards class-1. Figure 3 displays a box plot of class 1 tweets, which have a substantially lengthier content. The outliers present suggest that text length won't be a useful feature to consider.

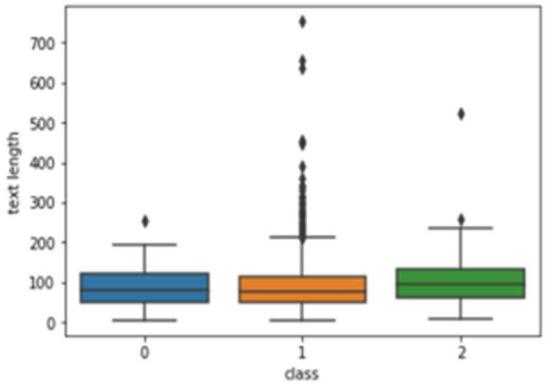

Figure 3. Box plot of class-1 tweets

C. *Data Pre-processing*

We use four processes to process the tweets: punctuation and capitalization removal, tokenization, stop word removal, and stemming. Splitting or tokenizing a sequence or text into a collection of tokens is known as tokenization. The process of reducing a word to its word stem, which attaches to prefixes and suffixes or to the roots of words known as a lemma, is referred to as stemming. [17-18] Here, the Porter Stemming Algorithm to locate the stem of the tweets.

D. *Feature Engineering*

Feature engineering is performed on the processed text, with features such as tf-idf weights, doc2vec vector columns, polarity of sentiment scores ,and other scores extracted and combined into multiple sets to fit into various classification models [19]. These models are then evaluated based on accuracy and f1-scores regarding different feature sets.

During the classification of Twitter text data, some features of tweets that can be used are:

- Tf-idf weights for n-grams: TF calculates the classic number of times a word is present within the text while IDF calculates the relative significance of this word depending on the number of texts the word can be found.

- Sentiment Analysis: We use Vader to determine the sentiment score of every text. It outputs 4 polarity scores: positive, negative, neutral and compound. It tells us not only about the positivity and negativity scores, but also about the degree to which an emotion is negative or positive [20].

- Doc2Vec Columns: Word2Vec works by transforming a word into a vector but Doc2Vec also combines all the words present in a sentence into a vector along with transforming a word into a vector. It does this by treating a sentence label as a unique 22 word and performs some operation on that special word. So, that special word becomes a label for a sentence.

Other enhancements twitter specific features include

a. Sum of syllables in the provided tweet
b. Length of the tweet text
c. Sum of words in a tweet
d. Sum of unique words
e. Average number of syllables
f. Readability metrics

We decided to perform the task of classification based on four features. These are tf-idf scores for n-gram range (1, 2), sentiment analysis scores, Doc2vec scores, and other enhanced features including readability scores. For simplification they are referred as F1= TF-IDF score, F2= TF-IDF with sentiment score, F3= TF-IDF with sentiment score and Doc2vec columns and F4= F3 and enhanced features.We also worked on further creating different feature sets based on the four features. They are F5= {TF-IDF scores, Doc2vec columns, other enhanced features}, F6= {TF-IDF scores, sentiment scores, other enhanced features} and F7= {sentiment scores, Doc2vec columns, other enhanced features}. These other feature sets were added to analyze the change in performance of the various classification algorithms that we use.

E. *Classifier Evaluation*

The performance of the developed classifier is evaluated using a range of performance measures. A few basic performance measures are briefly given here. F-Measure is calculated by taking the harmonic mean of precision and recall, with equal weightage for both variables. It allows a

model to be assessed using one score that accounts for both accuracy and recall, which is used for summarizing the model's performance and comparing models. [21]

$$\text{F-Measure} = \frac{2 \times (\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})} \quad (1)$$

Accuracy is the test data's percentage of correct predictions. It is computed by dividing the number of correct predictions by the total number of predictions. [22]-[24]

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \quad (2)$$

F. Model Comparison

We test these extracted features with four different machine learning algorithms. These algorithms are discussed briefly. Logistic Regression, one of the most common Machine Learning algorithms, is included in the Supervised Learning approach. It forecasts a categorical dependent variable based on a set of independent variables. The outcome must be either categorical or discrete. The Nave Bayes Classifiers are a group of Bayes' Theorem-based classification algorithms that all share a similar idea: each pair of characteristics being categorised is independent of the others. The purpose of the SVM Classifier is to determine the optimal line or decision boundary for categorising n-dimensional space into categories so that further data points can be placed in the correct category easily in the future. The Random Forest Classifier is a type of Supervised Machine Learning Algorithm that is commonly used in classification and regression problems. It generates decision trees from a variety of samples, employing the clear majority for classification and the mean for regression.

IV. Results And Analysis

The overall outcome of several analyses have been explained in this section. The accuracy and F-score of all analyses are shown in figure 4 and figure 5 respectively. The performance of various feature representation and classification algorithms in experimental contexts is shown. As seen in figure 4, naive bayes shows the least accuracy rate of 79.5% while random forest model for hate speech detection shows most promising performance with an accuracy of 92.7%.

F-score analysis is shown in figure 5. Here also, random forest generates the best f-score value of 91.8% while others like SVM, naive bayes and regression models recorded inferior performance as compared to random forest.

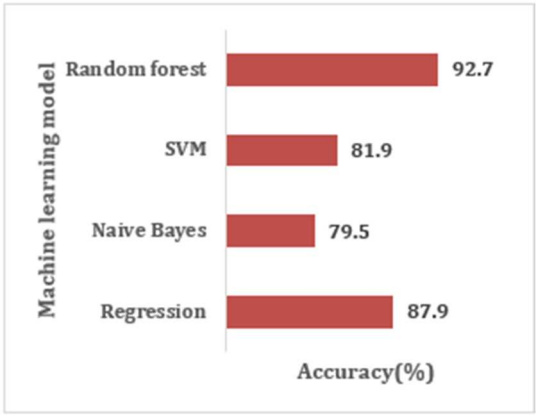

Figure 4. Accuracy analysis of the proposed work

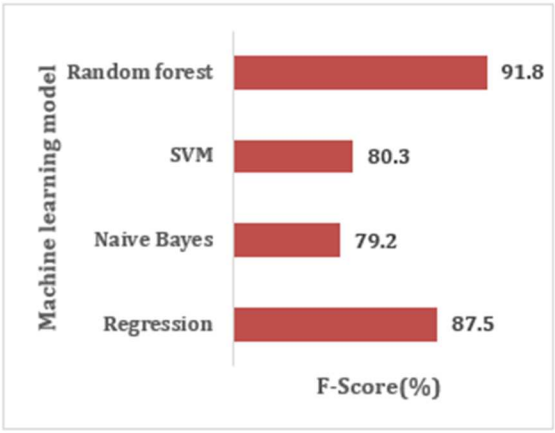

Figure 5. F-Score analysis of the proposed work

The Random Forest classifier performs well and exhibits noteworthy performance in different feature sets as shown in figure 6. Response delay generated by random forest is the least compared to others. The mean response delay using random forest model is only 2.1 seconds.

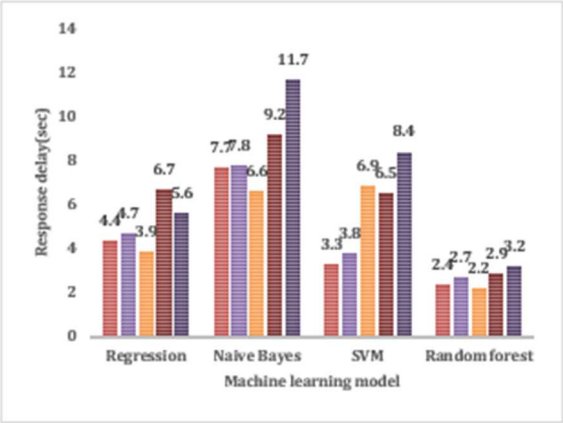

Figure 6. Response delay analysis of proposed model

We can see from the analysis that the most significant characteristic was tf-idf scores, which helps in improved hate speech detection. The sentiment scores can serve as a valuable tool for differentiating among hate speech and

obnoxious language. However, few other columns aren't found to be important in classification because they don't make much of a difference when they're eliminated from the feature set. When all of the graphs above are compared, Random Forest is the best-suited model.

V. Conclusion

To detect hate speech texts, this paper utilized text classification methods. Furthermore, to group hate speech, it compared multiple feature extraction techniques and four different machine learning algorithms. In comparison with Naïve Bayes theorem and SVM, Random Forest and Linear Regression algorithms produced better results. The lowest performance was seen by Naïve Bayes. With the rise of social media and hate speech it is significant that researchers develop an efficient model to monitor and eliminate the use of hate speech altogether.

References

[1] Tripathy, H. K., & Mishra, S. (2022). A Succinct Analytical Study of the Usability of Encryption Methods in Healthcare Data Security. In Next Generation Healthcare Informatics (pp. 105-120). Singapore: Springer Nature Singapore.

[2] Raghuwanshi, S., Singh, M., Rath, S., & Mishra, S. (2022). Prominent Cancer Risk Detection Using Ensemble Learning. In Cognitive Informatics and Soft Computing: Proceeding of CISC 2021 (pp. 677-689). Singapore: Springer Nature Singapore.

[3] Mukherjee, D., Raj, I., & Mishra, S. (2022). Song Recommendation Using Mood Detection with Xception Model. In Cognitive Informatics and Soft Computing: Proceeding of CISC 2021 (pp. 491-501). Singapore: Springer Nature Singapore.

[4] Sinha, K., Miranda, A. O., & Mishra, S. (2022). Real-Time Sign Language Translator. In Cognitive Informatics and Soft Computing: Proceeding of CISC 2021 (pp. 477-489). Singapore: Springer Nature Singapore

[5] Parikh, A., Desai, H., & Bisht, A.S. (2019): DA Master at HASOC 2019: Identification of Hate Speech using Machine Learning and Deep Learning approaches for social media posts. In: Proceedings of the 11th annual meeting of the Forum for Information Retrieval Evaluation (December 2019) CEUR-WS.org/Vol-2517/T3-1

[6] Gitari, N.D., et al., A lexicon-based approach for hate speech detection. International Journal of Multimedia and Ubiquitous Engineering, 2015. 10(4): p. 215-230

[7] Burnap, P. and M.L. Williams, Us and them: identifying cyber hate on Twitter across multiple protected characteristics. EPJ Data Science, 2016. 5(1): p. 11.

[8] Greevy, E. and A.F. Smeaton. Classifying racist texts using a support vector machine. in Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval. 2004.

[9] Kwok, I. and Y. Wang. Locate the hate: Detecting tweets against blacks. in the Twenty-seventh AAAI conference on artificial intelligence. 2013.

[10] Sharma, S., S. Agrawal, and M. Shrivastava, Degree based classification of harmful speech using twitter data. arXiv preprint arXiv:1806.04197, 2018.

[11] Nobata, C., et al. Abusive language detection in online user content. in Proceedings of the 25th international conference on the world wide web. 2016. International World Wide Web Conferences Steering Committee.

[12] Schmidt, A., & Wiegand, M. (2017, April). A survey on hate speech detection using natural language processing. In Proceedings of the Fifth International workshop on natural language processing for social media (pp. 1-10).

[13] Dinakar, K., R. Reichart, and H. Lieberman. Modeling the detection of textual cyberbullying. in fifth international AAAI conference on weblogs and social media. 2011.

[14] Liu, S. and T. Forss. Combining N-gram based Similarity Analysis with Sentiment Analysis in Web Content Classification. in KDIR. 2014.

[15] Juyal, V., Saggar, R., & Pandey, N. (2018). Clique-based socially-aware intelligent message forwarding scheme for delay tolerant network. International Journal of Communication Networks and Distributed Systems, 21(4), 547-559.

[16] Singh, A., Ali, M. A., Balamurugan, B., & Sharma, V. (2022, July). Blockchain: Tool for Controlling Ransomware through Pre-Encryption and Post-Encryption Behavior. In 2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT) (pp. 584-589). IEEE.

[17] Kokilavani, T., Gunapriya, D., Govindaraj, V., Pusphalatha, N., Hemalatha, N., Sharma, V., & Alkhayyat, A. (2023, February). Electric Vehicle Charging Station with Effective Energy Management, Integrating Renewable and Grid Power. In 2023 3rd International Conference on Innovative Practices in Technology and Management (ICIPTM) (pp. 1-5). IEEE.

[18] Gunapriya, D., Pusphalatha, N., Sudharsan, S., Pandi, S., Catherine, L., Sharma, V., & Alkhayyat, A. (2023, February). An Exhaustive Investigation of Battery Management System (BMS). In 2023 3rd International Conference on Innovative Practices in Technology and Management (ICIPTM) (pp. 1-5). IEEE.

[19] Ali, M. A., Balamurugan, B., Dhanaraj, R. K., & Sharma, V. (2022, November). IoT and Blockchain based Smart Agriculture Monitoring and Intelligence Security System. In 2022 3rd International Conference on Computation, Automation and Knowledge Management (ICCAKM) (pp. 1-7). IEEE.

[20] Singh, A., Dhanaraj, R. K., Ali, M. A., Balusamy, B., & Sharma, V. (2022, November). Blockchain Technology in Biometric Database System. In 2022 3rd International Conference on Computation, Automation and Knowledge Management (ICCAKM) (pp. 1-6). IEEE.

[21] Sharma, V., Balusamy, B., Thomas, J. J., & Atlas, L. G. (Eds.). (2023). Data Fabric Architectures: Web-Driven Applications. Walter de Gruyter GmbH & Co KG.

[22] M. N, R. D, S. Murali and V. Sharma, "Performance Analysis of DGA-Driven Botnets using Artificial Neural networks," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 2022, pp. 1-6, doi: 10.1109/ICRITO56286.2022.9965044.

[23] A. Maurya and V. Sharma, "Facial Emotion Recognition Using Keras and CNN", 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2022, pp. 2539-2543, doi: 10.1109/ICACITE53722.2022.9823480.

[24] Juyal, Vandana, Ravish Saggar, and Nitin Pandey. "An optimized trusted-cluster–based routing in disruption-tolerant network using experiential learning model. International Journal of Communication Systems 33.1 (2020): e4196.