Proactive Measures: Using Machine Learning to Anticipate and Combat Hate Speech on Twitter

By Akash Sawant,
Sy Mca Roll no.52158,div-B ,PES Modern College Of Engineering, Pune.

Prof. Yogeshchandra Puranik
Assistant Professor, MCA Department
P.E.S Modern College of Engineering

Abstract

This paper explores using machine learning (ML) to detect hate speech on Twitter. It collects a large dataset, preprocesses tweets, and extracts features. Various ML algorithms are trained and evaluated, including SVM, Naive Bayes, Logistic Regression, and Neural Networks. The study also investigates the impact of different features and explores techniques like word embeddings and deep learning. Results provide insights into effective methods for hate speech detection, crucial for combating online toxicity.

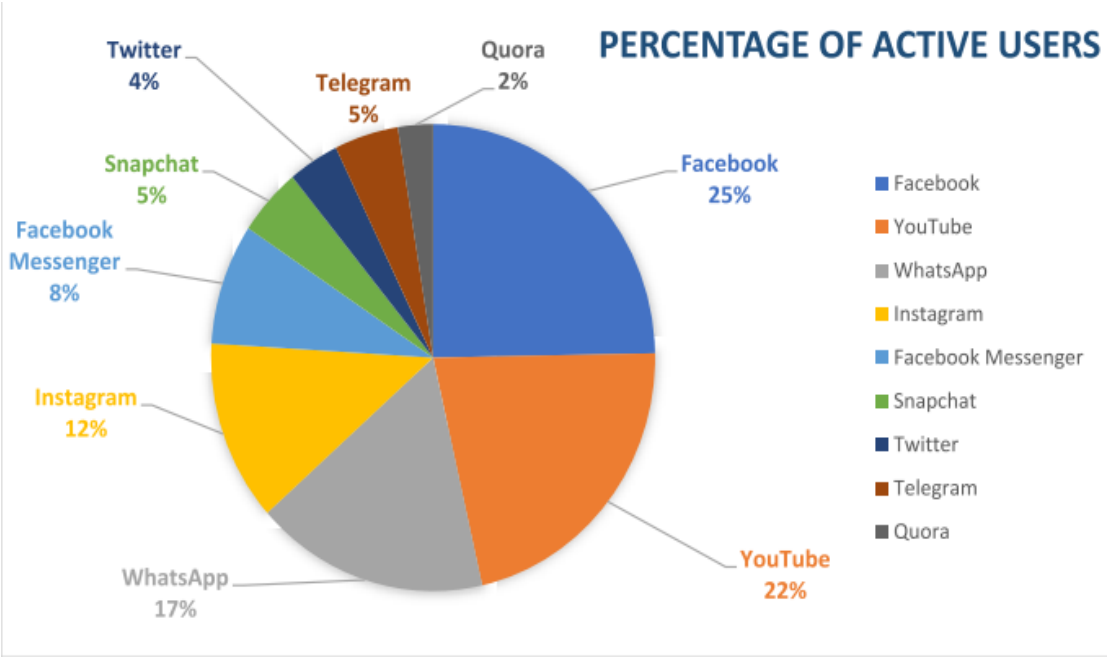
Keywords: Hate speech, Machine learning,Natural language processing, Social media

Introduction:

Social media platforms have become integral to modern communication, enabling individuals worldwide to connect, share ideas, and express opinions. However, alongside the benefits, these platforms have also witnessed the proliferation of hate speech, a phenomenon that poses serious threats to social cohesion, individual safety, and democratic discourse. Among these platforms, Twitter stands out as a prominent arena where hate speech often thrives due to its real-time nature, wide reach, and limited content moderation.

The prevalence of hate speech on Twitter underscores the urgent need for effective detection and mitigation strategies. Traditional methods of manually identifying and addressing hate speech are labor-intensive, time-consuming, and often ineffective in the face of the platform's scale. As such, the application of machine learning (ML) techniques for automated hate speech detection presents a promising avenue for addressing this pressing societal issue.

This research paper aims to explore and evaluate the efficacy of ML-based approaches for hate speech detection in Twitter. By leveraging the vast amounts of user-generated data available on the platform, coupled with advances in natural language processing (NLP) and ML algorithms, we seek to develop robust models capable of automatically identifying and classifying hate speech content.



Numerous research endeavors have delved into the realm of sentiment analysis and social media data for identifying and curbing cyber hate speech. This review examines several noteworthy studies in this domain, emphasizing the data scale, algorithmic methodologies, and resultant accuracies.

[1] Selma Ayşe Özel, Esra Saraç, Seyran Akdemir, Hülya Aksu et al. (2017) conducted a study targeting cyberbullying detection in Turkish texts via machine learning techniques. Utilizing data from Instagram and Twitter, they observed enhanced cyberbully detection by incorporating both words and emoticons as features. Their study identified Naïve Bayes Multinomial classifier as the top performer, achieving an 84% accuracy rate with feature selection.

[2] Ebraheem Fahad Aljarboua, Marina Bte Md. Din, Asmidar Abu Bakar et al. (2022) undertook a literature review exploring machine learning applications in cybercrime detection. Their findings underscored machine learning models' efficacy in identifying cybercrime, yielding accuracy rates spanning from 70% to 90%. The study aimed to compare various algorithms for automatic cybercrime detection, revealing Multiple layer perception model's supremacy with a 96% accuracy in existing data.

[3] Mifta Sintaha and Moin Mostakim conducted a study on machine learning algorithms' efficacy in detecting cyberbullying on social media. Leveraging a dataset of 2.5 million tweets, they trained two models: a naive Bayes model and a support vector machine model. Results favored SVM over Naive Bayes, with SVM achieving 89.54% accuracy in sentiment prediction, surpassing Naive Bayes' 73.03%.

[4] Barka Satya, Muhammad Hasan S J, Majid Rahardi, Ferian Fauzi Abdulloh et al. (2022) delved into sentiment analysis of Sestyc, a popular social media platform among Indonesian users. Employing text data from Google Play Store reviews, they aimed to gauge user sentiments toward Sestyc and identify the most effective sentiment classification algorithm. Their study incorporated Support Vector Machine, Logistic Regression, and Naive Bayes algorithms, with Support Vector Machine emerging as the most accurate, achieving an 87.81% accuracy rate out of 8,000 reviews.

Data Extraction:

The system utilizes the Twitter API and Python to gather a substantial volume of data from Twitter based on specified keywords, resulting in 118003 entries. The process involves importing necessary libraries, loading API credentials, authenticating, defining search parameters, extracting tweet texts, storing them in a list, converting to a Pandas DataFrame, and saving as a CSV file. This approach efficiently collects real-time tweets matching the keywords.

Preprocessing:

The preprocessing stage involves cleaning and transforming the collected tweet data for analysis. Steps include removing non-English tweets, converting text to lowercase, eliminating URLs, mentions, and hashtags, removing non-alphanumeric characters, tokenizing the text, filtering out non-English words and selected POS tags, removing stop words and "retweet", and lemmatizing words to their base form.

Data Folding:

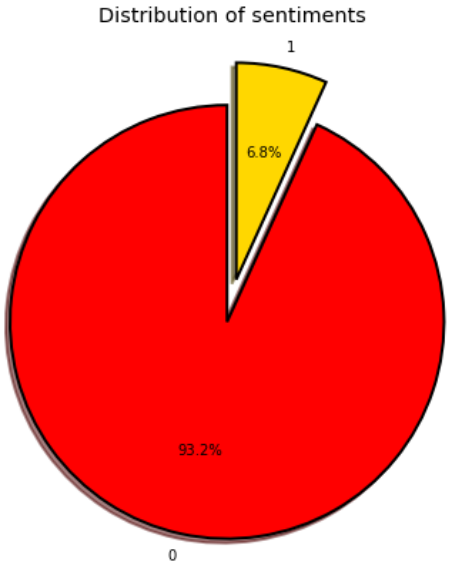
The system utilizes the data folding technique to split the dataset into 80% training and 20% testing sets using the train-test-split function from the sklearn-model-selection module, ensuring reproducibility with a random-state parameter set to 42.

Automated Training Set Classifier:

The code implements various classifiers including Naive Bayes (Multinomial NB), Logistic Regression, Convolutional Neural Network (CNN), and Support Vector Machine (SVM). Each classifier is trained on the preprocessed tweet data using the corresponding algorithm.

Evaluation:

The code classifies tweets into sentiment categories (Neutral, Positive, Negative, Very Positive, Very Negative) using the trained models. Each tweet is assigned a sentiment category based on its sentiment score or predicted label.



Feature Extraction:

The code employs the TF-IDF (Term Frequency-Inverse Document Frequency) vectorization technique for feature extraction. It utilizes the TfidfVectorizer class from scikit-learn to convert

preprocessed tweet text into numerical feature vectors. These extracted features serve as input for the classifiers.

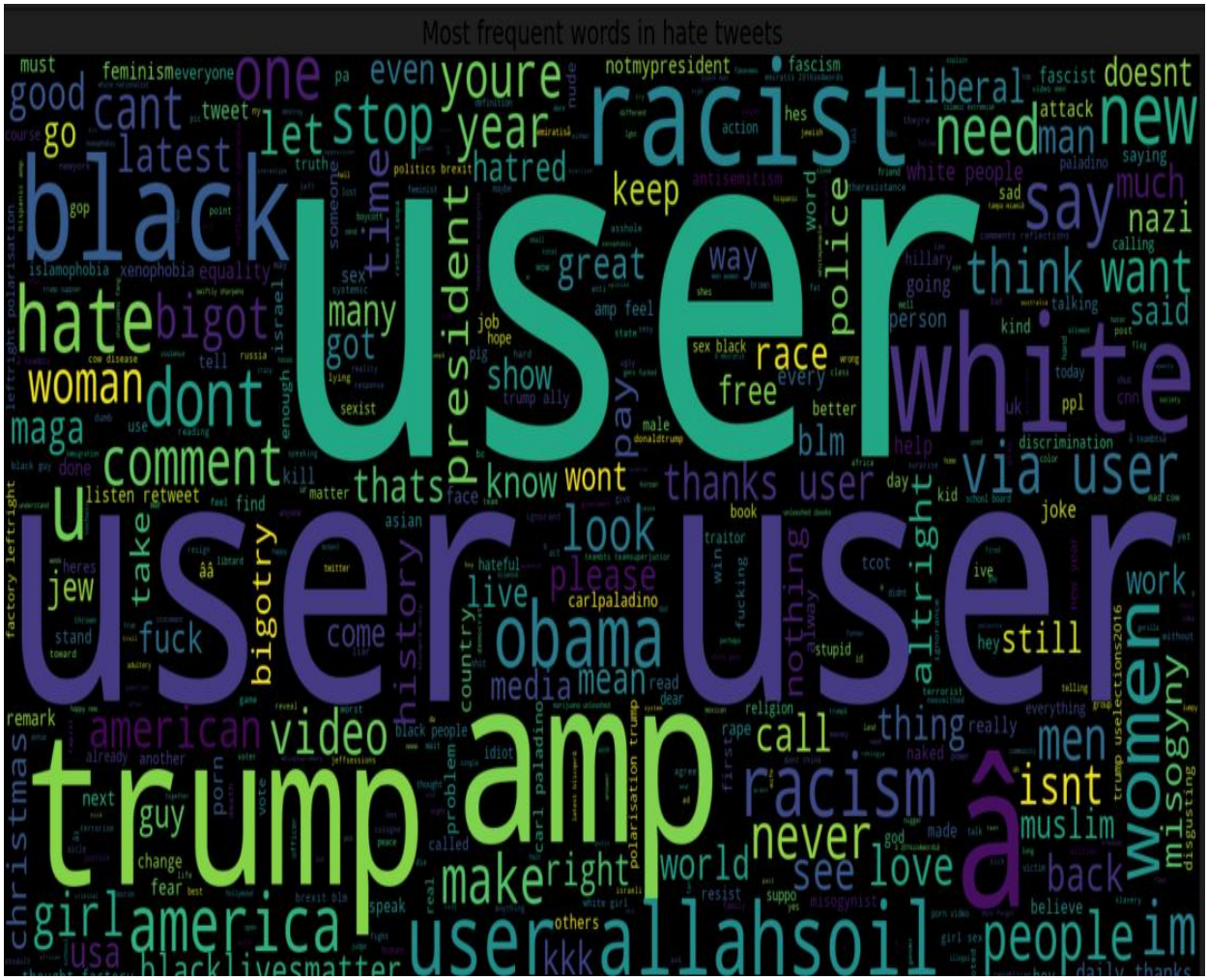


Fig. Most Frequent words in hate tweets

```
logreg = LogisticRegression()
logreg.fit(x_train, y_train)
logreg_predict = logreg.predict(x_test)
logreg_acc = accuracy_score(logreg_predict, y_test)
print("Test accuarcy: {:.2f}%".format(logreg_acc*100))
```

```
Test accuarcy: 93.17%
```

CONCLUSION

In conclusion, this project on hate speech detection in Twitter underscores the critical need for effective solutions to combat the proliferation of hate speech in social media platforms. Leveraging machine learning and natural language processing techniques, the project has explored diverse approaches, including Fuzzy classification, Random Forest, logistic regression, CNN-LSTM, and SVM.

The findings of the project emphasize the effectiveness of these machine learning approaches in detecting hate speech on Twitter. However, the accuracy of the models is contingent upon the quality of the training data and the algorithm's capacity to comprehend and analyze language effectively. The project's primary objective was to classify hate speeches using different models and identify the best-performing model with higher accuracy values.

Among the methods explored, the Fuzzy classifier emerges as a significant technique for hate speech classification, utilizing fuzzy hypergraphs and membership degree values to classify hate speech effectively.

Furthermore, the project underscores the importance of addressing hate speech on

social media platforms to ensure inclusivity and safety for all users. It highlights the potential of natural language processing and machine learning techniques in developing robust solutions to tackle hate speech.

Moving forward, future research will focus on enhancing the accuracy and robustness of these models by detecting various types of hate speech. In conclusion, this project serves as a testament to the importance of addressing hate speech in social media and showcases the potential of technology in developing effective solutions to mitigate this pervasive issue.

REFERENCES

[1] J. Cao, et al., A risky large group emergency decision-making method based on topic sentiment analysis, Expert Systems with Applications 195 (2022), 116527. [

2] P.K. Roy, et al., A framework for hate speech detection using deep convolutional neural network, IEEE Access 8 (2020) 204951–204962. [3] H. Liu, et al., A fuzzy approach to text classification with two-stage training for ambiguous instances, IEEE Transactions on Computational Social Systems 6 (2) (2019) 227–240.

[4] F.M. Plaza-Del-Arco, et al., A multi-task learning approach to hate speech detection leveraging sentiment analysis, IEEE Access 9 (2021) 112478–112489.

[5] S. Modha, et al., Detecting and visualizing hate speech in social media: A cyber watchdog for surveillance, Expert Systems with Applications 161 (2020), 113725.

[6] S. Kaur, S. Singh, S. Kaushal, Abusive content detection in online user-generated data: a survey, Procedia Computer Science 189 (2021) 274–281.

[7] F. Poletto, et al., Resources and benchmark corpora for hate speech detection: a systematic review, Language Resources and Evaluation 55 (2021) 477–523.

[8] M. Luo, X. Mu, Entity sentiment analysis in the news: A case study based on negative sentiment smoothing model (nssm), International Journal of Information Management Data Insights 2 (1) (2022), 100060.