Machine learning classifiers to detect offensive language in social media dataset

[1]Thirupurasundari.DR, [2]G. Abhinay Rao, [3]G. Sanjay, [4]G. Pranay, [5]E. Harinadh Reddy

[1]Associate Professor, [2,3,4,5] UG Student
Department of CSE
BIHER, Chennai.

***Abstract*- The uses of offensive languages on social media like Instagram, Facebook, Twitter, etc. are increased tremendously. Nowadays, huge growth in the number of social media users is seen. People are knowingly or unknowingly flooded the social media platforms with offensive posts. This became very challenging task to detect offensive posts. Manually solving this problem is not feasible, therefore automatic detection of offensive language is needed. Our objective of the work is to detect offensive language with the highest accuracy. The Davidson dataset is used for experiments with annotated posts and implemented based on ml methods. The proposed deep learning methods is compared with other known machine learning classifier. Overall, result analysis is show that proposed deep learning method is outperforming others.**

***Keywords*: Social media, offensive, Dataset, Classifier, Decision Tree, Machine Learning.**

INTRODUCTION

In today's digital age, the prevalence of offensive language online has become a significant concern. From social media platforms to comment sections, the spread of hateful and derogatory speech not only fosters toxicity but also poses serious threats to individuals' well-being and societal harmony. Addressing this issue requires robust systems capable of swiftly identifying and mitigating offensive content.

Machine Learning (ML) algorithms have emerged as powerful tools in this endeavor, offering the potential to automatically detect and filter out offensive language with high accuracy. By analyzing vast amounts of textual data, ML models can learn patterns and nuances inherent in offensive language, enabling them to effectively flag and categorize such content.

LITERATURE SURVEY

As increasing number of social media platforms have caused content overload. Unfortunately, not all contents are relevant, and some might harm people. This happen when there are people who misuse the platforms to propagate hate. Though difficult to achieve, identifying hate speech becomes an essential task in order to simultaneously provides freedom of speech and prevent hate speech content [1]. There are numerous studies focus on hate speech detection [2]. Theoretically, most of the theorist distinguish hate speech from merely offensive language [3]. Contrasting to the theorist, many studies overlook the offensive language. The studies focus more on the binary classification between hate speech and non-hate speech [4], [5] and fine-grained detection of various types of hate speech [6], [7]. Concatenate both classes, hate speech and offensive language has caused overset limits of hate speech that lead to false positive [8], [9]. The reputation of social media platforms can be tarnished when users feel frustrated as many non-hate speech contents are mistakenly detected as hate [10]. Although no formal definition of hate speech is universally been accepted, Fortuna and Nunes [11] has concluded the definition of hate speech from the numerous definitions as “language that attacks or diminishes, that incites violence or hate against groups, based on specific characteristics such as physical appearance, religion, descent, national or ethnic origin, sexual orientation, gender identity or other, and it can occur with different linguistic styles, even in subtle forms or when humor is used.” For example, the tweet “We agree... do you? http://t.co/4diz5NKYMN” F*CK YES, I DO! Send those illegal, wetbacks home!!!” is obviously contains racial insult towards immigrants as a target. Offensive language is “posts which are degrading, dehumanizing, insulting an individual, threatening with violent acts” [12]. As example, this tweet “and you look like hmmm a n*gger” contains offensive language content but does not incite violence or hatred to any groups based on specific characteristics. Nowadays, offensive words are commonly used in social media platforms. People tend to use offensive words in different context such as to express emotions like anger, frustration, or surprise [13]–[15]. For example: • Hate speech: "This n*ggah pharrell or whatever this n*ggahname is do not deserve sh*t for that white *ss g*y song" • Offensive language: “Lame n*ggaz wait in line, wait for b*tches, wait to create, wait for someone else to do sh*t for them. " • Neither: “I really need to take my rose coloured glasses off though. I gotta stop thinking everybody does sh*t with good intentions.” As shown in the example, offensive words “sh*t” can be found in all classes. The presence of offensive words in various aspects make detection becomes difficult. Although offensive words often considered rude and offensive or to emphasize emotion, offensive words should not to be taken lightly as the words can denote hate speech [16]. Other than offensive words, pejoratives appear in various classes as the use of “n*” pejorative in hate speech and offensive language in the previous example

1. *Scope and Impact of Offensive language*

In this paper, the literature survey underscores the interdisciplinary nature of research on offensive language detection, drawing insights from linguistics, computer science, psychology, and sociology. It highlights the ongoing efforts to develop robust, scalable, and ethical solutions for detecting and combating offensive language in online environments. Many studies emphasize the pervasive nature of offensive language in online platforms and its detrimental effects on individuals, communities, and society as a whole. They highlight the importance of developing effective detection mechanisms to mitigate its harmful impact.

2. *Datasets and Annotations*

Researchers have created annotated datasets of offensive language to facilitate the training and evaluation of machine learning models. These datasets typically include examples of hate speech, abusive language, and other forms of offensive content, labeled by human annotators. Various machine learning techniques, including traditional classifiers (e.g., SVM, Naive Bayes) and deep learning architectures (e.g., CNNs, RNNs), have been employed for offensive language detection. Feature engineering, such as word embeddings and syntactic features, plays a crucial role in capturing the semantic and contextual information of offensive language.

3. *Challenges and Limitations:*

The objective of this examine is to explore how machine learning algorithms can be utilized to automatically identify and filter out offensive content in digital platforms. The aim is to develop effective and accurate models that can distinguish between acceptable and offensive language, thereby fostering a safer and more respectful online environment. Additionally, the topic aims to discuss various methodologies, challenges, and ethical considerations involved in deploying such models, with the ultimate goal of promoting digital civility and combating online toxicity. Researchers acknowledge several challenges and limitations in the detection of offensive language, including the ambiguity of language, cultural and contextual variations, evolving forms of online abuse, and adversarial attacks aimed at circumventing detection systems.

4. *Applications and Real-World Deployments*

Finally, Offensive language detection systems have practical applications in content moderation, online safety measures, and platform governance. However, their deployment raises questions about privacy, censorship, and the balance between free speech and protection from harm.

EXISTING SYSTEM

In the existing system only one Machine Learning algorithm is used at once

The existing systems may integrate user feedback mechanisms to improve their performance over time. However, this integration is often limited compared to proposed systems. Existing systems typically have established architectures and models that undergo periodic updates and refinements. While these systems may receive enhancements to improve performance or address emerging challenges, their core functionalities remain relatively stable over time. Updates may include model retraining with new data, feature enhancements, or algorithmic improvements based on research advancements.

Disadvantages of Existing System

The Data Acquisition, Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. Time and Resources utilization is high.

PROPOSED SYSTEM

Our proposed approach uses machine learning filtering methods to detect the negative-related post from the social media data set. There are four main phases. In the first phase, social media text posts that relate to the crimes are extracted. In the second phase, we applied the preprocessing techniques to clean the data set. Then, we calculated then NLP values for each pre-processed post in the third phase. Finally, machine learning Based Filter is applied to remove non-related data. The Bagging classifier is used for classification to categorize the data.

Advantages of Proposed System

- Continuous Improvement: Machine Learning algorithms are capable of learning from the data we provide.
- The manual work is reduced and it helps do Automation for everything.
- It helps in the identification of the trends and patterns identification.
- It is useful for wide range of applications.

- Data Acquisition is simple.
- It is highly error-prone and gives the output very accurately
- The performance is very fast and accurate.

BLOCK DIAGRAM

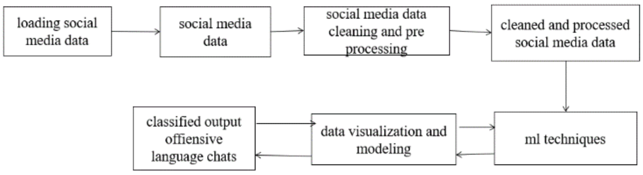

loading social media data
social media data
social media data cleaning and pre processing
cleaned and processed social media data
ml techniques
data visualization and modeling
classified output offiensive language chats

Fig-1: Block diagram of the software

Work Flow Diagram:

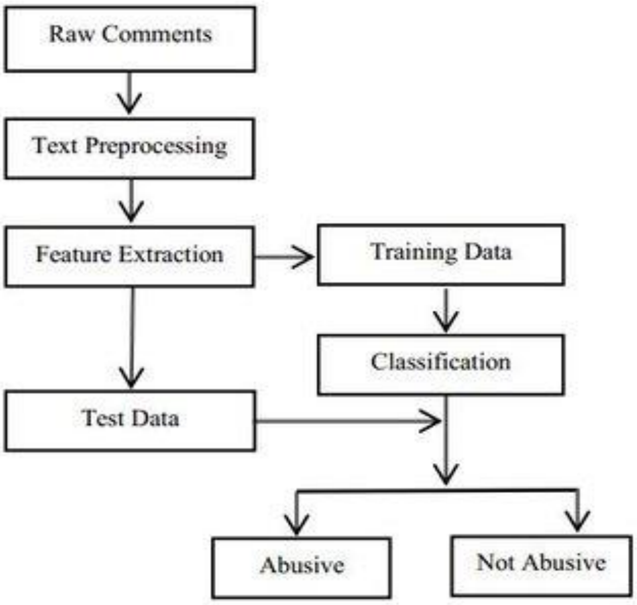


Fig-2: Work flow diagram

MODULES

1. Data collection

Data collection for detecting offensive language involves gathering textual data from online platforms like social media, forums, and chat rooms. Compliance with platform policies and ethical guidelines is essential. Techniques such as web scraping and filtering are used to collect and preprocess data. Annotation is performed to label instances of offensive language, ensuring quality through validation and augmentation. The process includes manual annotation by human annotators or automated methods. Data storage and management are crucial for organizing and protecting the collected data. Continuous monitoring and updates are necessary to adapt to evolving online discourse. The goal is to create a comprehensive dataset for training and evaluating machine learning models.

2. Data Cleaning

Data collection for detecting offensive language involves gathering textual data from online platforms like social media, forums, and chat rooms. Compliance with platform policies and ethical guidelines is essential. Techniques such as web scraping and filtering are used to collect and preprocess data. Annotation is performed to label instances of offensive language, ensuring quality through validation and augmentation. The process includes manual annotation by human annotators or automated methods. Data storage and management are crucial for organizing and protecting the collected data. Continuous monitoring and updates are necessary to adapt to evolving online discourse. The goal is to create a comprehensive dataset for training and evaluating machine learning models.

Data preaparation

It is common to find comments in which the same word or phrase is repeated consecutively. These repeated elements may add some noise to the tweet and give more importance to the same words that are not so important. We reduce the text and leave only the first appearance of the word or phrase. For example, the text ‘‘Correr es vivir, Correr es vivir, Correr es vivir, Correr es vivir’’ would be replaced by ‘‘Correr es vivir,’’

Feature Extraction

Feature extraction for offensive language detection involves converting textual data into numerical representations suitable for machine learning models. This process includes tokenization, lowercasing, and stopwords removal to standardize and preprocess the text. Vectorization techniques like one-hot encoding or word embeddings capture semantic relationships and contextual information. Additional features such as character n-grams and sentiment scores are engineered to capture linguistic characteristics and contextual cues. Dimensionality reduction techniques may be applied to enhance model efficiency by reducing the feature space. Overall, feature extraction facilitates the effective learning of patterns and relationships associated with offensive language, enabling accurate detection and classification.

Decision Tree- Decision tree classifiers are explicit factors and perform appropriately. Catch non-linearity. From interest pyspark. Ml. It is essential to demonstrate the likelihood of both predictive and modal judgment being imported because type is a Decision Tree Classifier**.**

Model training

BAG-OF-WORDS BASED MODELS The first type of models is based on extracting features from texts using BOW and classifying the texts using different Machine Learning methods. For this type of models, we pre-process texts applying stemming and removing stopwords. Additionally, we only keep terms with at least three occurrences in the collection and represent them by their TF-IDF scores in the collection (see Eq. 4). We have selected this representation after testing several variants, such as the use of lemmas, different categories of words, etc.

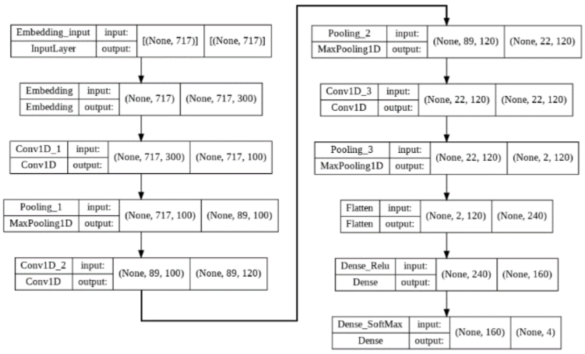


Fig-3: Decision Tree Dataset

RESULT AND DISCUSSION

Results and discussions in feature extraction for offensive language detection reveal the efficacy of various techniques in capturing linguistic nuances and contextual cues. Evaluation metrics such as accuracy, precision, recall, and F1-score demonstrate the performance of machine learning models trained on the extracted features. Comparative analyses highlight the strengths and limitations of different feature extraction methods, with some techniques proving more effective in certain contexts or datasets. Additionally, discussions delve into the interpretability of extracted features and their relevance to understanding offensive language patterns. Ethical considerations regarding bias, fairness, and privacy implications of feature extraction techniques are also addressed. Overall, the results and discussions underscore the importance of thoughtful feature selection and engineering in improving the performance and interpretability of offensive language detection systems

The Below graph depicts the accuracy of the machine learning models. Here in the graph we can see the SGD classifier is showing the least accuracy while the Bagging classifier is having the highest accuracy.

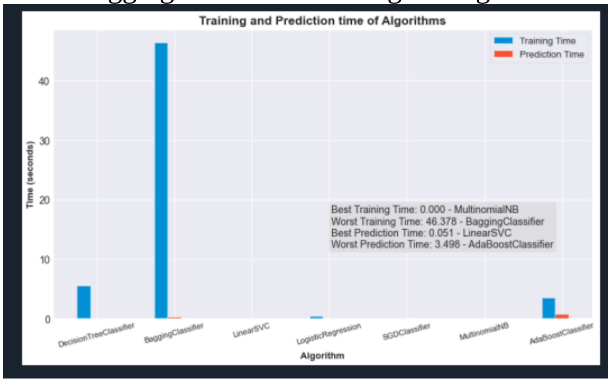
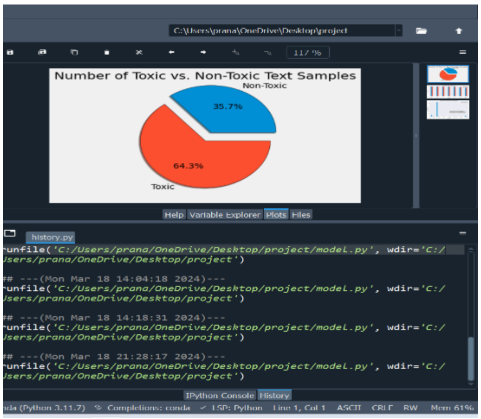


Fig-4: Training and Prediction time of algorithm

Model	Accuracy	precision
Decision Tree	0.9687	0.9543
Random Forest	0.9345	0.9141
KNeighbors Classifiers	0.8912	0.8112
Naïve Bayes	0.8586	0.8213

Table 1: Accuracy and precision using different Classifiers

Fig-5: Pie chart showing percentage of offensive language



```
runfile('C:/Users/prana/OneDrive/Desktop/project/model.py', wdir='C:/
Users/prana/OneDrive/Desktop/project')

## ---(Mon Mar 18 14:04:18 2024)---
runfile('C:/Users/prana/OneDrive/Desktop/project/model.py', wdir='C:/
Users/prana/OneDrive/Desktop/project')

## ---(Mon Mar 18 14:18:31 2024)---
runfile('C:/Users/prana/OneDrive/Desktop/project/model.py', wdir='C:/
Users/prana/OneDrive/Desktop/project')

## ---(Mon Mar 18 21:28:17 2024)---
runfile('C:/Users/prana/OneDrive/Desktop/project/model.py', wdir='C:/
Users/prana/OneDrive/Desktop/project')
```

```
nda (Python 3.11.7)   Completions: conda   LSP: Python   Line 1, Col 1   ASCII   CRLF   RW   Mem 61%
```

The above pie chart shows the information of the percentage of the presence of the foul or offensive words in the given dataset which are needed to be replaced.

EVALUATION METRICS

We evaluate our models using the common metrics found in the literature for evaluating the detection of offensive language:

Precision (see Eq. 1). It is the ratio between elements correctly classified as true instances, or true positives (TP), and all the instances classified as true (i.e. including the elements incorrectly classified as true instances, or false positives (FP)).

TP

(TP + FP)

(1)

Recall (see Eq. 2). It is the ratio between the TP and all the true elements (i.e. including the elements incorrectly classified as false instances, or false negatives (FN)).

TP

(TP + FN)

(2)

F1-score (see Eq. 3). This metric combines precision and recall using the harmonic mean.

F1 = 2 ∗

(precision ∗ recall)

(precision + recall)

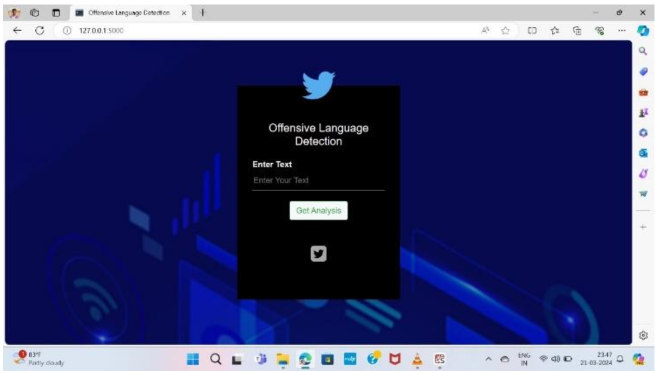


Fig-6: Final frontend output of the website

CONCLUSION

The In this paper, we 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. We performed comparative analysis of Logistic Regression, Naive Bayes and Support Vector Machines on various sets of feature values and model hyperparameters. The results showed that Logistic Regression performs better with the optimal n-gram 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.

FUTURE ENHANCEMENT

Developing efficient algorithms and models capable of real-time offensive language detection is important for applications such as content moderation and online chat filtering. Future work could focus on optimizing models for low latency and high throughput deployment.

Ensuring the privacy of users' data is crucial in offensive language detection systems, especially in applications where user-generated content is analyzed. Future research could explore privacy-preserving techniques such as federated learning or differential privacy.

Investigating methods to make offensive language detection models more robust against adversarial attacks is important. Adversarial attacks can include intentionally crafted input to evade detection systems. Developing techniques to defend against such attacks is essential for real-world deployment

REFERENCES:

1. Ivey-Stephenson, A.Z.; Demissie, Z.; Crosby, A.E.; Stone, D.M.; Gaylor, E.; Wilkins, N.; Lowry, R.; Brown, M. Suicidal Ideationand Behaviors Among High School Students—Youth Risk Behavior Survey, United States, 2019. MMWR Suppl. 2020, 69, 47–55.[CrossRef] [PubMed]
2. Gliatto, M.F.; Rai, A.K. Evaluation and Treatment of Patients with Suicidal Ideation. Am. Fam. Physician 1999, 59, 1500. [PubMed]
3. Giachanou, A.; Crestani, F. Like it or not: A survey of Twitter sentiment analysis methods. ACM Comput. Surv. 2016, 49, 1–41.[CrossRef]
4. Oussous, A.; Benjelloun, F.-Z.; Lahcen, A.A.; Belfkih, S. ASA: A framework for Arabic sentiment analysis. J. Inf. Sci. 2019, 46, 544–559.[CrossRef]
5. Pachouly, S.J.; Raut, G.; Bute, K.; Tambe, R.; Bhavsar, S.; Students, U. Depression Detection on Social Media Network (Twitter) using Sentiment Analysis. Int. Res. J. Eng. Technol. 2021, 8, 1834–1839. Available online: www.irjet.net (accessed on 23 April 2022).
6. Waseem, Zeerak, and Dirk Hovy. "Hateful symbols or hateful people? predictive features for hate speech detection on twitter." Proceedings of the NAACL student research workshop. 2016.
7. Davidson, Thomas, et al. "Automated hate speech detection and the problem of offensive language." Proceedings of the International AAAI Conference on Web and Social Media. Vol. 11. No. 1. 2017.
8. Fortuna, Paula, et al. "A survey on automatic detection of hate speech in text." ACM Computing Surveys (CSUR) 52.4 (2019): 1-30.
9. Zhang, Meeyoung, et al. "Detecting offensive language in social media to protect adolescent online safety." Information Sciences 441 (2018): 138-152.
10. Nobata, Chikashi, et al. "Abusive language detection in online user content." Proceedings of the 25th International Conference on World Wide Web. 2016.
11. Malmasi, Shervin, and Marcos Zampieri. "Detecting hate speech in social media." Proceedings of the First Workshop on NLP and Computational Social Science. 2016.
12. Burnap, Pete, and Matthew L. Williams. "Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making." Policy & Internet 9.2 (2017): 223-242.
13. Chatzakou, Despoina, et al. "Detecting cyberbullying and cyberaggression in social media." Internet Research 30.1 (2020): 211-230.
14. Zannettou, Savvas, et al. "What is Gab: A Bastion of Free Speech or an Alt-Right Echo Chamber?" Proceedings of the International AAAI Conference on Web and Social Media. Vol. 13. No. 01. 2019.
15. Qian, Jie, et al. "Hierarchical recurrent neural network for offensive language detection in online comments." Information Processing & Management 57.6 (2020): 102279.