Threat Detection using Machine Learning

Name: Vu Trong Chau
Student ID: 1687347
Email: vchau@troy.edu

***Abstract*—The rapid advancement of technology, particularly through social media platforms such as Facebook, TikTok, Instagram, and YouTube, has transformed the way people interact and communicate. While these platforms offer numerous benefits, they also expose users to significant challenges and risks. Among the most concerning issues for young people are humiliation, insults, bullying, and harassment, which may originate from strangers, anonymous users, or even friends. Cyberbullying is defined as a deliberate act intended to intimidate or distress others, resulting in feelings of fear, discomfort, and vulnerability.**

To tackle the issue of cyberbullying, this study proposes the development of an advanced recognition system designed to identify and classify instances of cyberbullying. This system employs a comprehensive feature extraction method that encompasses various types of features, including semantic features that analyze the meaning of words, sentimental features that assess the emotional tone of messages, syntactic features that examine sentence structure, and pragmatic features that consider context and usage. To further validate the effectiveness of the proposed system, I compare the performance of several machine learning algorithms using different extracted feature sets, including Multinomial Naive Bayes, Logistic Regression, Random Forest, Decision Tree, and XGBoost. The experimental results show promising performance in terms of classification metrics. The findings highlight the importance of integrating diverse features to enhance the detection and recognition of cyberbullying behavior, providing valuable insights for improving the functionality of cyberbullying recognition systems.

***Index Terms*— Threat Detection, Machine Learning, Text Classification, Logistic Regression, XGBoost, Multinomial Naive Bayes, Random Forest, Decision Tree, ROC Curve, AUC Score.**

I. INTRODUCTION

A. *Background*

In recent years, the rise of smart mobile devices has intensified bullying, particularly among adolescents. Bullying is defined as the malicious use of electronic communication tools to threaten, harass, or humiliate someone. This troubling behavior can take many forms and is commonly found on popular platforms, including social networking sites, online gaming communities, chat rooms, and discussion forums. It encompasses a range of actions beyond the typical tactics of hacking someone's account or impersonating them online. Bullying can include posting offensive comments, spreading misinformation, and engaging in targeted harassment, all of which can significantly harm a person's reputation and self-esteem.

Determining the prevalence of bullying and identifying its victims can be challenging. The impact of bullying can lead to considerable emotional distress. Victims often experience anxiety, depression, feelings of isolation, and in severe cases, suicidal thoughts. Many individuals feel ashamed or afraid to seek help, making it difficult for friends, family members, or authorities to recognize their struggles. A study conducted in the United States involving nearly 4,000 students in grades 6 to 8 found that, within the previous two months, 11% of students reported being victims of bullying, 4% admitted to being bullies, and 7% identified as both bullies and victims. Despite increasing awareness of the social implications of bullying, there is still insufficient intervention and attention to the issue. There is an urgent need for innovative solutions and frameworks that can foster a safer and more equitable online environment, ensuring that young users can engage in digital spaces without the threat of harassment and intimidation [1].

B. *Motivation*

Social media platforms are essential communication tools and important sources of information for researchers. These platforms enable users to share and receive information instantaneously, often in the form of breaking news before it appears in mainstream media. Given their high user activity and large audience, social media has become a significant resource for analyzing bullying and its prevalence in society. Cyberbullying may be more prevalent than traditional bullying. While traditional bullying often occurs within schools and families, providing some respite for victims, cyberbullying can reach individuals anytime and anywhere, exposing them to a potentially vast audience. This form of bullying is complex and can profoundly impact the lives of its victims. The negative effects can be extensive, leading to issues such as low self-esteem, poor academic performance, and various emotional disorders. Many victims of bullying report experiencing serious negative consequences. In severe cases, psychological distress may result in self-harm or suicidal behavior. These alarming statistics highlight the severity of cyberbullying and the urgent need to address its consequences. Identifying and intervening in cases of cyberbullying is crucial. By raising awareness and taking decisive action, we can help protect children's mental health and ensure their safety online [1].

C. Problem Statement

The primary goal of this project is to develop a robust model for threat detection using advanced machine learning techniques, leveraging public datasets. To achieve this, we will extract several sentences from the text, highlighting words related to bullying by bolding them and displaying a percentage that indicates the level of bullying present in the paragraph. This thorough analysis aims not only to demonstrate the effectiveness of these features in detecting online bullying but also to emphasize their importance in identifying harmful behaviors in discussions. By comparing the performance of various algorithms, the project seeks to determine which technique yields the best results for addressing this significant issue.

II. LITERATURE REVIEW

A. Overview of Threat Detection

In recent years, bullying has become a prominent topic in the media, often linked to several high-profile teen suicides. Detecting threats in the context of cyberbullying—which includes aggressive or abusive online communication capable of causing emotional or psychological harm—can be quite complex. Cyberbullying often involves repetitive actions conducted through electronic means that aim to inflict psychological distress. The challenges in detection are compounded by the diverse and often difficult-to-manage language used on social media platforms. This language can include sarcasm, phrases, or slang, making traditional keyword detection methods less effective. A Canadian study published in 2010 surveyed over 2,000 students from grades 6 to 12, revealing that 25% had experienced a cyberbullying incident in the three months prior. Additionally, 8% reported engaging in cyberbullying, while another 25% identified themselves as victims. These numbers have likely increased with the rise of technology today [1].

To address this issue, this project aims to develop a machine learning system that analyzes text content to identify signs of cyberbullying. The detection process involves several stages: preprocessing, feature extraction, and classification. Preprocessing eliminates noise such as emojis, stop words, and special characters. During feature extraction, we convert text into numerical representations using methods like TF-IDF and vectorization. The focus is on employing advanced deep learning techniques to train and evaluate various algorithms, including Multinomial Naive Bayes, Logistic Regression, Random Forest, Decision Tree, and XGBoost. The system is designed not only to classify input text as either bullying or non-bullying but also to highlight abusive language and provide a percentage score indicating the severity of the bullying. Ultimately, this system aims to serve as both a diagnostic and preventive tool, contributing to a safer online environment. This approach enhances the accuracy and explainability of threat detection models, enabling stakeholders, such as educators, parents, and moderators, to intervene early and effectively.

B. Logistic Regression

Logistic Regression is a supervised machine learning algorithm for estimating the probability of an instance belonging to a particular class. It is a statistical technique that examines the relationship between two variables. In this post, we will discuss the fundamentals of logistic regression, its types, and its implementation. Logistic regression is used primarily for binary classification. It employs the sigmoid function with independent variables as input and yields a probability value ranging between 0 and 1. The equation will be [2]:

$$p(X; b, \omega) = \frac{e^{\omega.X+b}}{1 + e^{\omega.X+b}} = \frac{1}{1 + e^{-\omega.X+b}}$$

For y = 1, predicted probabilities will be: $p(X; b, \omega) = p(x)$

For y = 0, predicted probabilities will be: $1 - p(X; b, \omega) = 1 - p(x)$

C. Multinomial Naive Bayes

Multinomial Naive Bayes is a type of Naive Bayes algorithm primarily used for classification tasks. It is founded on the Theorem of Bayes and is especially suited for discrete data, commonly applied in text classification. This algorithm uses word frequencies as counts and assumes that all features or words follow a multinomial distribution. Multinomial Naive Bayes is notably popular for applications such as classifying documents, particularly in spam email detection based on word frequencies. The equation is used to calculate the probability of a message belonging to a certain category:

$$P(X) = \frac{n!}{n_1!\, n_2! \ldots n_m!} p_1^{n_1} p_2^{n_2} \ldots p_m^{n_m}$$

n represents the total number of trials, n_i indicates the number of occurrences of outcome i, and p_i denotes the probability of outcome i [3].

D. Random Forest

Random Forest is a machine learning algorithm that provides more accurate predictions by utilizing multiple decision trees. Each tree is trained on different random subsets of the data, and their results are combined using voting for classification tasks and averaging for regression tasks. This approach leads to improved accuracy and reduced errors [4].

E. Decision Tree

A Decision Tree is a tool that helps us make decisions by listing various alternatives along with their possible outcomes. It is commonly used in machine learning for both prediction and classification tasks. It provides a hierarchical representation in a tree-like structure. It begins with a single central question, referred to as the root node, which represents the entire dataset [5].

F. XGBoost

XGBoost stands for eXtreme Gradient Boosting, a powerful machine learning algorithm known for its efficiency, speed, and high performance. It is an enhanced version of Gradient Boosting and falls under the category of ensemble learning techniques, which leverage a collection of weak models to create a stronger model. XGBoost primarily uses decision trees as its base learners and improves performance by sequentially combining individual models. Each new tree learns from the errors of the previous tree, which is the essence of the boosting process. Moreover, XGBoost enables model building through parallel processing, allowing users to work with large datasets swiftly. Additionally, XGBoost offers user-defined customization, permitting adjustments to model parameters to maximize performance based on the specific problem at hand. The model can be represented as:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i)$$

$\hat{y}_i$ is the final predicted value for the i^{th} data point, K is the number of trees in the ensemble, and $f_k(x_i)$ represents the prediction of the K^{th} tree for the i^{th} data point [6].

III. PROPOSED SYSTEM

The proposed system aims to identify and categorize instances of bullying in online written texts, with a particular focus on occurrences of malicious or offensive language. By using a hybrid approach that combines high-power feature extraction methodologies with machine learning algorithms, the system accurately detects threatening behavior by analyzing the significance and composition of the texts. It visually highlights offensive words and provides a percentage indicating the potential threat level for easier interpretation.

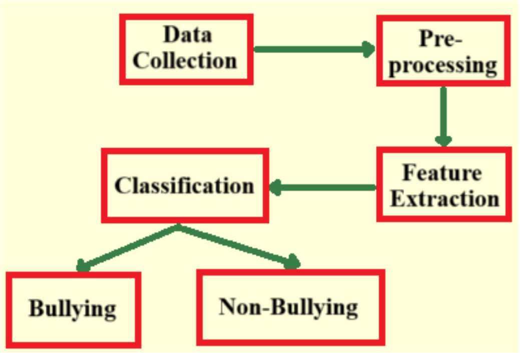

Fig. 1. Proposed Model Framework

The system begins with a preprocessing stage, where the raw social media data is cleaned and normalized. This stage involves removing user information, duplicates, repetitive personalities, links, and any unused columns. Following this, encoding and normalization processes are applied to prepare the data for analysis.

After preprocessing, the text undergoes a feature extraction stage. Here, the multidimensional aspects of the language are analyzed. Semantic features capture meaning through techniques like TF-IDF and word vector representation, helping to determine the contextual significance of words. Sentiment features assess the emotional tone of the message, classifying it as positive, negative, or neutral. Syntactic features analyze sentence structure, punctuation, and word types, while pragmatic features examine context, intent, and social cues embedded in the language.

These extracted features are then fed into various machine learning algorithms, including Multinomial Naive Bayes, Logistic Regression, Decision Tree, Random Forest, and XGBoost. Each model is trained and tested with different combinations of feature sets to identify which method produces the highest accuracy and confidence in threat detection.

The final step of the system generates an output indicating whether bullying is present and provides a bullying severity score. This score is calculated based on the intensity and likelihood of bullying, reflecting the classifier's confidence level and the relevant linguistic features.

IV. METHODOLOGY

A. Data Collection

The dataset used in this study is a structured CSV file containing two main columns: one for text content and another for binary values, where 0 indicates non-bullying and 1 indicates bullying. The dataset comprises 72,471 sentences, which are used for training and testing machine learning models. Preprocessing the data is essential to ensure that the models are evaluated accurately based on the dataset. Class balance is a critical factor in measuring classification performance metrics such as accuracy, precision, recall, F1-score, and AUC.

Since raw text data must be converted into a numeric format suitable for machine learning algorithms, we employed the Term Frequency-Inverse Document Frequency (TF-IDF) vectorization process. This method highlights the relative importance of words in each title compared to the entire dataset, thereby minimizing the influence of common words and enhancing the significance of more discriminative keywords. Additionally, the titles underwent preprocessing using standard natural language processing (NLP) techniques, including converting to lowercase, removing punctuation, and tokenization. These processes normalize the data, reduce noise, and improve feature extraction and classification accuracy.

TABLE I. DATASET

Number of Data	Non-Bullying	Bullying
72,471	52,868	19,603

B. Data Processing

Data preprocessing involves the removal of user data, duplicate entries, unnecessary characters, URLs, and irrelevant columns. Once this initial cleanup is completed, coding and normalization are utilized to prepare the data for analysis. When dealing with real-world text or social media posts in natural language processing, it is common to encounter irrelevant content or characters. For instance, numbers and punctuation often do not contribute to the detection of bullying and should be removed before using the data with machine learning models. This can be accomplished through various preprocessing steps, including eliminating unwanted characters such as stop words, punctuation marks, and numbers, as well as quoting and parsing the original material.

Parsing refers to breaking down a volume of text into words, phrases, or other meaningful units. After coding and removing stop words, stemming is applied to transform words back to their root forms, enhancing the model's capacity to recognize patterns despite grammatical variations. Finally, the cleaned and processed tokens are converted into numeric representations using encoding methods like TF-IDF (Term Frequency-Inverse Document Frequency). These techniques transform the text into vectors that machine learning models can process. Normalization or scaling techniques can also be employed to standardize the features.

C. Performance Evaluation

To evaluate the performance of different models, we used multiple metrics to ensure a comprehensive evaluation including accuracy, precision, recall, F1 score, and AUC. These metrics are widely recognized in fields such as data mining and support systems, making them essential for evaluating model performance and ensuring reliable results.

- *F-1 Score*

F1 Score is an important metric in machine learning, providing a balanced measure of the precision and recall of a model. The F1 Score formula is derived from the harmonic mean of precision and recall, making it an essential component in the F1 Score, Precision, and Recall framework. This metric is especially useful in situations where the class distribution is unbalanced [7].

$$Accuracy = \frac{correct\ classifications}{total\ classifications}$$

An ideal model would achieve an accuracy of 100%, meaning it would have zero false positives and zero false negatives. However, a model that predicts negative 100% of the time may appear to have high accuracy up to 99% in highly imbalanced datasets where one class occurs very infrequently, such as only 1% of the time. Despite this high accuracy, the model would not be useful. 1% of the time. Despite this high accuracy, the model would not be useful [8] [9].

- *Precision*

Precision is the proportion of all positive classifications of the model that are positive. It is defined mathematically as follows:

$$Precision = \frac{correctly\ classified\ actual\ positives}{everything\ classified\ as\ positive}$$

A hypothetical perfect model would have no false positives and thus have an accuracy of 1.0 [8].

- *Recall*

Recall is the proportion of all actual positive results that are correctly classified as positive. It is defined mathematically as follows:

$$Recall = \frac{correctly\ classified\ actual\ positives}{all\ actual\ positives}$$

A hypothetical perfect model would have zero false negatives and thus have a recall rate of 1.0, wich is a detection rate of 100%. In an imbalanced dataset, where the number of true positives is very low, the recall rate is less meaningful and less useful as a metric [8] [9].

- *Accuracy*

Accuracy is the proportion of all correct classifications, whether positive or negative. It is defined mathematically as follows:

$$\text{Accuracy} = \frac{\text{correct classifications}}{\text{total classification}}$$

A perfect model would have zero false positives and zero false negatives and therefore an accuracy of 1.0, or 100% [8].

- *Area under the curve (AUC)*

The area under the ROC curve (AUC) indicates the likelihood that a randomly selected positive example will be ranked higher than a randomly selected negative example by the model. For a perfect model, represented by a square with sides of length 1, the AUC is 1.0. This means that there is a 100% chance that a randomly selected positive example will be correctly ranked higher than a randomly selected negative example. In other words, regardless of where the threshold is set, the AUC reflects the probability that the model will rank a randomly selected positive example (represented by a square) higher than a randomly selected negative example (represented by a circle), based on the distribution of data points [9].

- *True Positive Rate (TPR)*

The True Positive Rate measures the number of actual positive cases that the model correctly identifies. It is defined as:

$$TPR = \frac{correctly\ classified\ actual\ positives}{all\ actual\ positives}$$

In situations with unbalanced datasets where there are very few true positives, recall is a more important metric than accuracy. Recall assesses the model's ability to correctly detect all positive instances, which is essential in applications like disease prediction. The consequences of a false negative are often more serious than those of a false positive. For a specific example illustrating the difference between recall and accuracy metrics, please refer to the notes in the recall definition [8].

- *False Positive Rate (FPR)*

The False Positive Rate (FPR), also known as the probability of false alarm, refers to the percentage of all real negatives that are incorrectly identified as positives. It is defined as follows:

$$FPR = \frac{incorrectly\ classified\ actual\ negatives}{all\ actual\ negatives}$$

In an ideal scenario, a model would achieve a false alarm rate of 0%, meaning there would be no false positives, resulting in an FPR of 0.0. However, FPR becomes less meaningful and useful as a metric in unbalanced datasets, particularly when there are very few actual negatives—let's say just 1 or 2 examples in total [8].

V. RESULT

To provide a thorough overview of the threat detection data, we employ detailed summary statistics for key metrics. We also include a pie chart that clearly illustrates the percentage of bullying cases compared to non-bullying cases, enabling a precise and comprehensive analysis of the data.

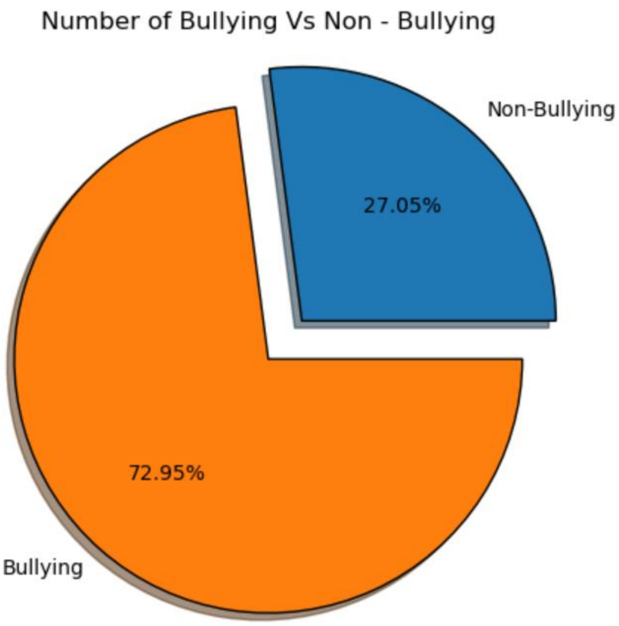

Fig. 2. Overview of Threat Detection Data

The pie chart unequivocally displays the distribution of bullying versus non-bullying cases in the dataset. It reveals that a significant 72.95% of the data is classified as bullying, while only 27.05% is classified as non-bullying. These datasets encompass a broad range of contexts for social media and online communication. The language used is often informal and conversational, featuring abbreviations and slang. This informal tone can complicate analysis, and it's crucial to recognize that many users tend to make common spelling mistakes. Understanding these factors is essential for the accurate interpretation of the data.

A. Comparision Metrics

As shown in the table below, RNN is the best model compared to other models in absolute percentage, which shows that it is highly computationally accurate. CNN performs slightly worse than RNN, but it still shows that it is a good model and computationally capable to a certain extent.

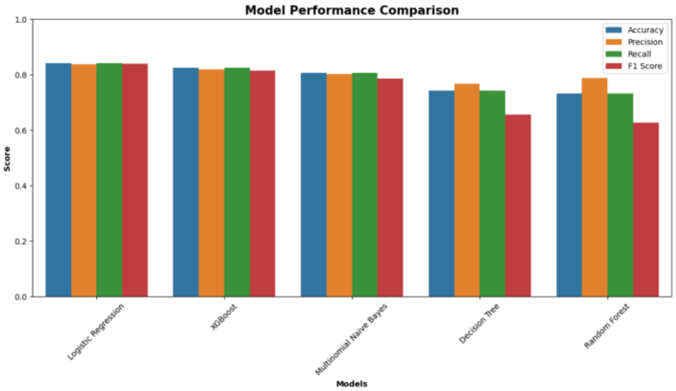

Fig. 3. Bar Chart of Model Performance

TABLE II. COMPARISION METRICS TABLE

Data Models	Accuracy	Precision	Recall	F1-Score	AUC
Logistic Regression	84.18%	83.87%	84.18%	83.98%	91.02%
XGBoost	82.64%	81.97%	82.64%	81.60%	91.11%
Multinomial Naive Bayes	80.68%	80.20%	80.68%	78.53%	88.54%
Decision Tree	74.34%	76.79%	74.34%	65.56%	78.36%
Random Forest	73.26%	78.91%	73.26%	62.67%	86.85%

- *Logistic Regression*

The most reliable and well-balanced model in this study is Logistic Regression. It demonstrated consistency across other important metrics, including precision (83.87%), recall (84.18%), and F1-score (83.98%), achieving the highest overall accuracy of 84.18%. These results indicate that the model was effective in accurately identifying both instances of bullying and non-bullying. Additionally, even with a highly imbalanced dataset, it achieved an AUC score of 91.02%, reflecting a strong ability to differentiate between the two classes. Furthermore, the interpretability of logistic regression is a significant advantage, making it suitable for applications where explainability and transparency are essential.

- *Multinomial Naive Bayes*

The Multinomial Naive Bayes model demonstrated respectable performance with an accuracy of 80.68%, a precision of 80.20%, a recall of 80.68%, and an F1-score of 78.53%. It performed competently and achieved a good balance across all metrics, although it did not outperform XGBoost or Logistic Regression. While its AUC of 88.54% is not as strong as the top two models, it still indicates that Naive Bayes can effectively distinguish between classes. What sets Naive Bayes apart is its simplicity, speed, and efficiency, making it an attractive option for lightweight applications or resource-constrained environments.

- *Random Forest*

The Random Forest model performed the worst among those tested. It recorded the lowest accuracy (73.26%), recall (73.26%), and F1-score (62.67%), despite achieving a relatively high precision of 78.91%. This discrepancy indicates that Random Forest failed to identify many actual bullying cases, as highlighted by its low F1-score, even though it had fewer false positives when detecting bullying messages. Furthermore, its AUC of 86.85% was lower than that of the Naive Bayes, XGBoost, and Logistic Regression models. These results suggest that Random Forest may have overfitted to the majority class, a common issue when dealing with unbalanced datasets that have not been properly adjusted.

- *Decision Tree*

Decision Tree model achieved an accuracy of 74.34%, precision of 76.79%, and recall of 74.34%. However, its F1-score was noticeably lower at 65.56%, indicating that the model performed poorly overall. The decline in F1-score suggests that while the model is relatively good at detecting bullying messages (as shown by the precision), it struggles to maintain a healthy balance between false positives and false negatives. Furthermore, its AUC (Area Under the Curve) of 78.36% indicates a limited ability to differentiate between bullying and non-bullying messages. Although decision trees are easy to understand and intuitive, they are prone to overfitting, especially with datasets that contain imbalances like this one.

- *XGBoost*

XGBoost is quite similar to Logistic Regression and even surpasses it in certain areas. Among all the models tested, XGBoost achieved the highest accuracy at 82.64% and an AUC of 91.11%. This suggests that XGBoost performs particularly well in ranking the probabilities of bullying versus non-bullying. Although its precision (81.97%), recall (82.64%), and F1-score (81.60%) are slightly lower than those of Logistic Regression, the differences are negligible. Given its ability to handle complex relationships and feature interactions effectively, XGBoost is an excellent choice for production systems that require top notch performance.

B. *Receiver Operating Characteristic (ROC) Curve*

The above graph provides a detailed comparison of the performance of three separate machine learning models: Multinomial Naive Bayes, Logistic Regression, Random Forest, Decision Tree, and XGBoost. On the x-axis, various evaluation metrics are listed, including Accuracy, Precision, Recall, and F1 Score. The y-axis shows the corresponding values for each metric, allowing for easy visual evaluation and comparison.

ROC curve analysis provides a graphical and numerical summary of each model's performance in distinguishing between non-bullying and bullying content across all classification levels. Among all the models evaluated, XGBoost achieved the highest AUC (0.9111), indicating its superior ability to identify true positives while minimizing false positives. Logistic Regression closely followed with an AUC value of 0.9102, demonstrating nearly identical discriminative ability, along with the added benefits of interpretability and sparsity.

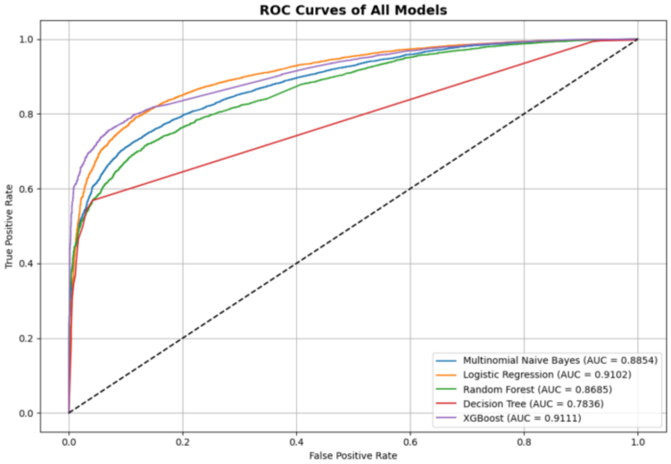

Fig. 4. ROC Curve Comparison

Multinomial Naive Bayes also performed well, achieving an AUC of 0.8854, making it a safe and lightweight option for rapid predictions. In contrast, Random Forest and Decision Tree exhibited lower performance, with AUCs of 0.8685 and 0.7836, respectively. This may be due to the class imbalance present in the dataset, which impaired their ability to effectively differentiate between the classes. XGBoost and Logistic Regression emerged as the best-performing models based on their ROC curve analyses, highlighting their strong potential for implementation in real-world bullying detection systems.

VI. DISCUSSION

A. *Analysis of Results*

After evaluating five machine learning models, it was found that XGBoost and Logistic Regression outperformed the others in detecting cyberbullying. Among them, Logistic Regression achieved the highest accuracy along with commendable precision, recall, and F1 score. On the other hand, XGBoost recorded the highest AUC score of 91.11%, indicating its strong ability to differentiate between bullying and non-bullying posts. While XGBoost's overall classification scores were slightly lower than those of Logistic Regression, its boosting strategy and capacity to recognize complex patterns made it a more suitable choice for this application.

The Decision Tree and Random Forest models did not perform as well, particularly in recall and F1 score, which highlighted their limitations in detecting all instances of bullying. Although Multinomial Naive Bayes is relatively simple, it demonstrated good accuracy and AUC, making it an appropriate option for lighter usage. These results further confirm that a combination of thorough preprocessing, diverse feature extraction, and robust algorithms significantly enhances threat detection. Additionally, the system's ability to highlight abusive words and provide severity scores for bullying actions improves interpretability and facilitates its application in educational, social, and moderation contexts.

B. *Practical Applications*

The findings of this comparative analysis of various machine learning models provide valuable insights for real-world applications, particularly in the field of threat detection. Given the vast amount of user-generated content on social media platforms, it is crucial to use precise and effective classification models for the automatic detection of malicious activity. Models like XGBoost and Logistic Regression, which demonstrate high AUC and strong overall performance metrics, are well-suited for integration into content moderation systems. They can provide moderators with real-time alerts for review. Additionally, these models have the potential to be adapted for use in various domains due to their robust performance. This analysis can assist developers and organizations in selecting the model that best meets their unique requirements and constraints.

C. *Limitations*

The limitation in this study is that the dataset used restricts the scope of the analysis. The included texts lack comprehensive information and are insufficiently detailed for thorough analysis. This lack of data hinders the system's ability to provide deeper insights into user behavior and patterns of social media interaction. However, we believe that this study serves as an important starting point for researchers and practitioners looking to assess the prevalence of bullying across various contexts. Our findings can serve as a foundation for developing a multi-level machine learning classifier that can better categorize and analyze instances of bullying.

D. *Future Scope*

I plan to explore a broader range of deep learning and machine learning methods for potential future research opportunities in threat detection. Our approach will include utilizing larger datasets and implementing various feature extraction techniques, such as social features, network features, and user characteristics. Furthermore, we believe that using an ensemble model could lead to more accurate predictions. This approach combines the outputs of several baseline models to create a final prediction. Our goal is to enable users to observe comparable levels of bullying severity across social media platforms by developing an ensemble model that integrates multiple algorithms or distinct datasets from various well-known social media networks.

VII. CONCLUSION

This study used five different classification models for the bullying identification task to evaluate the performance of each model. The analysis was based on several performance metrics, including reliability and accuracy values. The optimal model for each project depends on the specific needs of the application. Based on the evaluation of performance, XGBoost emerged as the most effective choice for threat detection systems, providing a powerful combination of features.

For future research in threat detection, we plan to explore additional machine learning techniques, utilize larger datasets, implement more advanced feature extraction methods, and conduct more in-depth system inference. This may involve examining features related to the user, network, and social context.

REFERENCES

[1] Peebles, E. (2014). Cyberbullying: Hiding behind the screen. *Paediatrics & Child Health*, *19*(10), 527–528. https://doi.org/10.1093/pch/19.10.527

[2] GeeksforGeeks. (2025, June 3). *Logistic regression in machine learning*. GeeksforGeeks. https://www.geeksforgeeks.org/machine-learning/understanding-logistic-regression/

[3] GeeksforGeeks. (2025a, May 30). *Multinomial naive Bayes*. GeeksforGeeks. https://www.geeksforgeeks.org/machine-learning/multinomial-naive-bayes/

[4] GeeksforGeeks. (2025c, June 27). *Random Forest algorithm in machine learning*. GeeksforGeeks. https://www.geeksforgeeks.org/machine-learning/random-forest-algorithm-in-machine-learning/

[5] GeeksforGeeks. (2025d, June 30). *Decision tree*. GeeksforGeeks. https://www.geeksforgeeks.org/machine-learning/decision-tree/

[6] GeeksforGeeks. (2025a, May 23). *XGBoost*. GeeksforGeeks. https://www.geeksforgeeks.org/machine-learning/xgboost/

[7] GeeksforGeeks. (2025f, July 15). *Evaluation metrics in machine learning*. GeeksforGeeks. https://www.geeksforgeeks.org/machine-learning/metrics-for-machine-learning-model/

[8] *Classification: Accuracy, recall, precision, and related metrics*. (n.d.). Google for Developers. https://developers.google.com/machine-learning/crash-course/classification/accuracy-precision-recall

[9] GeeksforGeeks. (2025c, June 3). *F1 score in machine learning*. GeeksforGeeks. https://www.geeksforgeeks.org/machine-learning/f1-score-in-machine-learning/

Dataset

[10] *Tweets dataset for detection of Cyber-Trolls*. (2018, July 12). Kaggle. https://www.kaggle.com/datasets/dataturks/dataset-for-detection-of-cybertrolls?resource=download

[11] *Cyberbullying dataset*. (2022, October 22). Kaggle. https://www.kaggle.com/datasets/saurabhshahane/cyberbullying-dataset

[12] *Cyberbullying Classification*. (2022, January 17). Kaggle. https://www.kaggle.com/datasets/andrewmvd/cyberbullying-classification