Detection of Cyberbullying in Social Media Texts using Explainable Artificial Intelligence

by

Mohammad Rafsun Islam

A thesis submitted to the
School of Computing
in conformity with the requirements for
the degree of Master of Science

Queen's University
Kingston, Ontario, Canada
May 2023

Abstract

The widespread use of social media has opened the door to new forms of harassment and abuse, such as cyberbullying, that have a serious impact on individuals' psychological health, especially children and teenagers. Therefore, research communities have recently paid attention to developing detection approaches using Natural Language Processing (NLP) combined with machine learning algorithms to identify instances of cyberbullying in social media texts such as comments, posts, and messages. Those approaches have successfully classified the social media text as either cyberbullying or non-cyberbullying. However, they are unable to determine the type of cyberbullying and the reasons why victims may be targeted based on certain characteristics. The aim of this thesis is to develop a novel detection approach that can identify the type of cyberbullying based on characteristics such as gender, religion, age, and ethnicity. This thesis has accomplished this objective by utilizing an Explainable Artificial Intelligence (XAI) technology called Local Interpretable Model-agnostic Explanations (LIME) to justify and explain the classification of text as cyberbullying. LIME enables machine learning models to capture and highlight the most influential words that affect the decision to classify a text as cyberbullying. Those influential words are utilized to re-label and update the training data. The machine learning models are then re-trained using the updated data. To evaluate the

performance of the proposed approach, a simulation experiment has been conducted on a large dataset containing comments and posts from Twitter. Simulation results show that: 1) LIME provides reliable and convincing justifications and explanations for classifying a text as cyberbullying; 2) LIME enables machine learning to identify the type of cyberbullying based on characteristics such as gender, religion, age, and ethnicity; and 3) LIME improves the performance of the machine learning models in terms of classification accuracy.

Acknowledgments

I would like to express my sincere appreciation to my supervisor, Dr. Mohammad Zulkernine, for his continuous guidance, support, and encouragement throughout my Masters at Queen's University. His expertise, mentorship, and patience have significantly contributed to the completion of this thesis. I am also grateful to Dr. Ahmed Bataineh for his insightful feedback, constructive criticism, and continuous assistance during my research journey.

Finally, I would like to thank my parents and my wife for their unwavering support throughout my academic journey. Their love and support have been a source of strength and motivation, without which this accomplishment would not have been possible.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Social media has revolutionized the way we communicate and interact with others. With the advent of various social media platforms, people can now connect with others from all over the world, share their thoughts and ideas, and build relationships like never before. Social media has become an integral part of modern life, transforming the way we consume news, express ourselves, do business, and entertain ourselves. However, this widespread use of social media has also given rise to new forms of harassment and abuse that significantly affect mental health, such as cyberbullying.

Cyberbullying is a form of online harassment that happens through various offensive acts, including sending vulgar messages, posting inappropriate and hurtful comments, propagating online rumors, posting inappropriate images and videos, and cyberstalking [1, 2]. With the increasing use of social media and other online platforms, cyberbullying has become a pervasive issue that affects people of all ages, especially children and teenagers. The effects of cyberbullying can be devastating, leading to psychological distress, social isolation, and even suicide. Studies [3, 4]

indicate that cyberbullying affects over 60% of US and 35% of Canadian teenagers. The Pew Research Center found that 59% of US adolescents who are victims of cyberbullying show at least one symptom of stress, embarrassment, isolation, and aggression [4]. According to a study published by US News [5], 7.6% of US children who experience cyberbullying consider suicide. As a result, cyberbullying has become a serious problem that requires urgent attention.

1.2 Research gaps and objectives

The computer science research community is paying attention to developing detection approaches that can identify instances of cyberbullying in text posted on social media platforms such as comments and posts. Mainly, Natural Language Processing (NLP), along with other machine learning algorithms, is utilized to learn, analyze, understand, and identify patterns of harassing or abusive behavior in social media texts [6, 7, 8]. However, the current capabilities of these approaches are limited to classifying social media texts into two categories: cyberbullying and non-cyberbullying, which we refer to in this thesis as "binary classification". Specifically, binary classification is unable to determine the type of cyberbullying and the reasons why victims of cyberbullying may be targeted based on certain characteristics, such as gender, religion, age, and ethnicity. Such information are needed by both psychologists and social media platforms to appropriately assist the victims and build proactive strategies that may help prevent cyberbullying attacks from occurring in advance. For instance, social media platforms can leverage such information to prevent bullies who engage in abusive behavior against a certain race from seeing posts made by individuals belonging to that race. Furthermore, these approaches are unable to provide

justifications and explanations for their decisions to classify a text as cyberbullying or not. Such information can assist programmers in diagnosing their learning models and identifying underlying technical errors to improve model performance. In addition, the performance of these approaches in terms of classification accuracy is sometimes unsatisfactory, with relatively low accuracy.

This thesis aims to address the following research gaps:

- Firstly, we aim to develop a novel detection approach that can identify the type of cyberbullying based on characteristics such as gender, religion, age, and ethnicity. This type of classification is referred to as "Multiclass classification" in this thesis.

- Secondly, we aim to enable the proposed approach to provide justification for why a text is classified as a certain type of cyberbullying. Specifically, the approach will be designed to highlight the most influential factors that affect the classification decision.

- Lastly, we aim to improve the classification accuracy of the proposed approach.

1.3 Proposed Approach Overview

This section outlines the proposed approach for detecting instances of cyberbullying in social media texts. The approach, depicted in Figure 1.1, comprises three layers: Text Pre-processing, Text Training, and Decision Explanation.

In the layer of Texts Pre-processing, Natural Language Processing (NLP) technology is utilized to prepare the data (social media texts) for the training process. Specifically, we use three NLP tools; text preprocessing, sentiment analysis, and

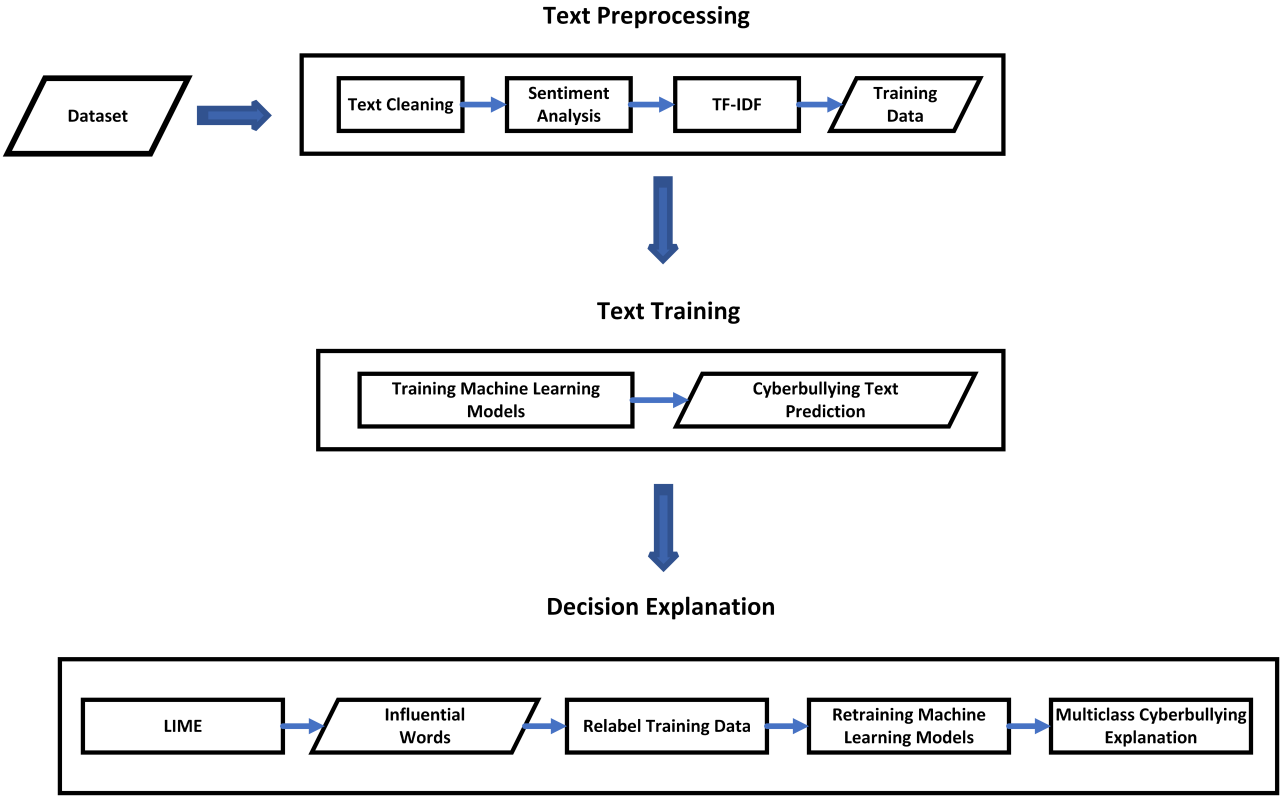


Figure 1.1: Overview of Cyberbullying Detection Model

Term Frequency-Inverse Document Frequency (TF-IDF). Text preprocessing removes noises, such as punctuation, from social media texts. Sentiment analysis is employed to identify the sentiment degree of the social media texts. Term Frequency-Inverse Document Frequency (TF-IDF) is employed to identify the importance score for each word in a social media text [9]. The words' importance scores help identify offensive words and consequently detect cyberbullying in social media texts. The output training data is reshaped into records where each record includes a social media text, TF-IDF score for each word in the text, sentiment score for the text, and finally, the record is labeled as cyberbullying or non-cyberbullying according to those scores.

In the second layer, Text Training, four different machine learning models are

trained on the data generated by the first layer (Text Pre-Processing). Those machine learning models, including Random Forest Classifier, Gradient Boosting, Support Vector Machine (SVM), and Long Short-Term Memory (LSTM), have been selected due to their high efficiency in prediction and classifying texts in the state of the art of Artificial intelligence and Machine Learning technologies. Given a social media text as an input, each of the four machine learning models in the second layer identifies whether the input text contains cyberbullying or not. Those models work in parallel and independently in the sense that they are not sharing inputs and outputs with each other. However, the objective behind utilizing four models then is to conduct further practical experiments to conclude the most efficient technique for this task.

In the third layer, Decision Explanation, Explainable Artificial Intelligence (XAI) technology is utilized to justify the decisions taken by the four machine learning models in the second layer. Mainly, we use the Local Interpretable Model Agnostic Explanation (LIME), one of XAI's algorithms, to capture the most influential words on the classification of social media texts as cyberbullying or not. The weights of those influential words, namely by explanations, highlight the reasons why victims of cyberbullying may be attacked based on certain characteristics, such as gender, religion, age, and ethnicity. The training dataset generated by the first layer is then re-labeled by the explanations generated by the third layer. Specifically, if it contains cyberbullying, each text (a record) in the updated training dataset is re-labeled by one of the following classes: gender, religion, age, and ethnicity. We then re-train the machine learning models for multiclass cyberbullying predictions. Finally, we utilize LIME to generate explanations for the multiclass cyberbullying predictions for highlighting the influential multiclass cyberbullying words and comparing the

performances of machine learning models.

1.4 Contributions

The thesis focuses on developing a novel approach for cyberbullying detection that has three main contributions.

- Firstly, the proposed approach identifies the type of cyberbullying by utilizing characteristics such as gender, religion, age, and ethnicity. This approach is referred to as "Multiclass classification" in the thesis, which is a substantial advancement compared to existing methods that only classify cyberbullying into binary categories. Multiclass classification enables the detection of different types of cyberbullying, which can vary in severity, intent, and impact. This approach provides a more comprehensive understanding of cyberbullying and can lead to better prevention and intervention strategies.

- Secondly, the proposed approach not only classifies a text as a particular type of cyberbullying but also provides justification for the classification decision. Specifically, the approach highlights the most influential factors contributing to the classification decision. This feature is essential because it allows for a better understanding and interpretation of the results, which can help improve the model's performance. The justification can also assist in explaining the classification outcome to stakeholders such as educators, parents, and law enforcement officials, enabling them to take appropriate action. By providing justification for classification decisions, the proposed approach enhances the transparency and reliability of the cyberbullying detection system.

- Lastly, the approach improves the overall classification accuracy.

Those contributions are accomplished through the application of LIME, which assists in identifying the most relevant features for the classification process. LIME provides interpretability and transparency by highlighting the most significant factors that substantially contribute to classifying a text as a specific type of cyberbullying. This involves identifying the most influential words and other pertinent characteristics such as gender, religion, age, and ethnicity. The explanations generated by LIME are utilized to re-label and update the training data. The machine learning models are then re-trained using the updated data. This step improves the classification accuracy of the machine learning models and achieves the objective of executing multiclass classification. The ultimate goal of this research is to develop a robust and accurate cyberbullying detection system that can be effectively used in real-world applications. The application of LIME enhances the reliability and interpretability of the system, which are critical for building trust among stakeholders.

1.5 Thesis Organization

The thesis is structured and organized as follows. Chapter 2 provides a literature review of ongoing research on detecting cyberbullying in social media texts. In Chapter 3, we present our proposed approach to identifying the type of cyberbullying using NLP, machine learning models, and XAI technology. In Chapter 4, we discuss the simulation results of the proposed approach. Finally, Chapter 5 concludes the thesis and suggests future directions for research in cyberbullying detection.

Chapter 2

Related Work

In this chapter, we discuss the current state of the solutions proposed to detect cyberbullying in social media texts. These solutions are categorized and organized in this chapter based on the utilized AI and machine learning technologies, which include Natural Language Processing (NLP), Random Forest Classifier (RFC), Gradient Boosting, Support Vector Machine (SVM), Long Short-Term Memory (LSTM), and Explainable Artificial Intelligence (XAI).

2.1 Natural Language Processing (NLP)

Natural Language Processing (NLP) technology has emerged as one of the most significant techniques employed by research communities due to its efficiency in identifying the degree of cyberbullying in texts. Typically, research proposals make use of the algorithms of sentiment analysis and Term Frequency-Inverse Document Frequency (TF-IDF) to identify the emotional undertone of a text and extract keywords from its content [10]. The sentiment analysis algorithm relies on collecting and training a massive amount of data, including individuals' feelings about a wide range of topics discussed on social media platforms [11]. The results of the training process are

categorized and saved in a dictionary known as a lexicon dictionary. The lexicon dictionary contains the polarity scores for words to determine whether a word is positive, negative, or neutral [11]. For example, the word 'stupid' has a score of -1 and is considered a negative word.

Several proposals have used sentiment analysis to identify cyberbullying in social media texts. For instance, Xu et al. [12] employed sentiment analysis to distinguish the emotions and feelings expressed in the posts of victims and bullies. They observed that victims often convey negative emotions like depression and loneliness, whereas bullies typically express angry emotions. Harsh Dani et al. [13] conducted an empirical study to evaluate the sentiment distribution among ordinary and cyberbullying posts. They found that cyberbullying posts have significantly higher negative sentiment scores than other posts. Nahar et al. [10] developed an algorithm that matches bullies with victims by analyzing the hyperlink structure of online social networks. They then calculate sentiment scores for both predators and victims to rank the most influential individuals.

Term Frequency-Inverse Document Frequency (TF-IDF) is a statistical method used in NLP to determine the relevance of a word in a text [14]. A word's TF-IDF score typically reflects its frequency among social media texts or how common it is [15]. Research studies, such as [16, 17, 8], use the TF-IDF technique to identify and rank key offensive words by importance. Dinakar et al. [14] employed TF-IDF to detect commonly used profane and stereotypical words in a YouTube dataset. They found that most offensive words are ethnic slurs for different races of people. Chia et al. [18] used the TF-IDF technique to identify sarcasm and irony in a dataset of tweets. Wu et al. [19] used an improved version of TF-IDF to identify the positions

of offensive words in texts. They found that offensive words usually appear at the beginning or end of texts.

In this thesis, popular NLP tools such as sentiment analysis and TF-IDF are used to detect cyberbullying texts on social media sites like Twitter. These tools are used to generate sentiment and TF-IDF scores for each word in a text and combine them to identify the offensive words that contribute the most to the negative sentiment of a cyberbullying text.

2.2 Random Forest Classifier (RFC)

Random Forest Classifier (RFC) is a popular ensemble machine learning model that leverages the collective power of multiple individual models to improve overall predictive performance [20]. Typically, RFC involves constructing multiple decision trees and integrating their outcomes to produce a final prediction [21]. Specifically, the final prediction of RFC is made by combining the predictions of individual decision trees and selecting the most frequent outcome through majority voting [20]. This model aims to enhance performance while minimizing overfitting data, which is commonly observed in decision trees. The RFC model is widely used in various applications, including classification and regression tasks, and has demonstrated high accuracy and robustness in many real-world scenarios.

RFC has been widely used to develop predictive models for identifying instances of cyberbullying in text. For instance, Novalita et al. [21] utilized RFC model to detect cyberbullying instances in Indonesian tweets. The authors preprocessed the training dataset to include new features such as frequencies of offensive words and the feelings associated with words. Their research achieved an impressive F-1 score of 0.90,

indicating a high accuracy level in detecting cyberbullying. Squicciarini et al. [22] employed RFC to detect users who engage in bullying behavior on social media networks. The authors trained their model using the information on the user's activities, including posts and comments, as well as personal details such as age, gender, and social networks of their friends. The results of their study demonstrate that the RFC approach achieved an accuracy rate of 83.9%. Despoina et al. [23] applied a similar approach to identify a bully on Twitter. Their results achieved a precision of 89.9% and a recall of 91.7%. Al Garadi et al. [1] utilized three feature selection algorithms: the chi-square test, information gain, and Pearson correlation, to extract features from tweets for identifying cyberbullying content on Twitter. These features included the user's online presence, age, gender, vulgar words, and anger-related words. The authors then employed four machine learning algorithms, including Naive Bayes, SVM, RFC, and KNN, to predict cyberbullying tweets on social media. Their research indicates that the RFC model achieved high precision and recall results. Specifically, the precision of the RFC algorithm was 90%, and the recall was 94%. Balakrishnan et al. [24] employed Random Forest along with two other classifiers, Naive Bayes and J48, to develop an automated cyberbullying detection model for classifying cyberbullying texts into the following categories: normal, bully, aggressor, and spammer. The authors applied their model to a dataset consisting of 5453 tweets and achieved an accuracy of 91%.

The Random Forest algorithm has consistently shown promising results in detecting instances of cyberbullying on social media. Its ability to effectively handle both numerical and categorical features makes it particularly suitable for tasks involving the identification of cyberbullying, which often involve a combination of textual

content, user behavior, and social media characteristics. Due to these reasons, the Random Forest model is employed in this research for predicting cyberbullying texts.

2.3 Gradient Boosting

Gradient Boosting is a machine learning algorithm that repeatedly trains a sequence of weak decision trees to reduce prediction errors and create a stronger predictive model. The algorithm typically begins by creating a single decision tree that is trained on the dataset. The model then calculates the residuals, which are the differences between the predicted and actual values of the target variable for all the records in the dataset. These residuals are incorporated into the training data. The algorithm creates and trains a new decision tree on the updated training data, focusing on the records with high residuals to minimize them. This process is repeated until the algorithm reaches the minimum residual value [25, 26].

Multiple proposals have suggested using the Gradient Boosting model to identify cyberbullying instances in social media texts. For instance, Kurniawanda et al. [27] utilized the Extreme Gradient Boosting model (XGBoost) to detect cyberbullying content in Instagram comments. Their study achieved an accuracy score of 75.2%. Muneer et al. [26] employed the Light Gradient Boosting model alongside other machine learning models, including Random Forest Classifier, Logistic Regression, Naive Bayes, Adaboost, Stochastic Gradient Descent, and SVM, to create an ensemble machine learning model for identifying cyberbullying texts in a dataset of 37,373 tweets. Their model achieved a high accuracy rate of 90% in identifying cyberbullying texts correctly. In their study, Alam et al. [23] developed an ensemble machine learning model by integrating and combining Gradient Boosting with Adaboost and Bagging

Classifier to detect and classify cyberbullying content. The authors initially employed TF-IDF to extract additional features, such as the frequency of offensive words and then used these feature values along with the social media text dataset to train their ensemble model. The implemented model achieved a high accuracy rate of 96%.

Gradient Boosting is used in this research since it is widely recognized for its high predictive capabilities and often outperforms other machine learning algorithms in various classification problems. In the context of cyberbullying text prediction, a Gradient Boosting model is particularly useful as it can achieve high accuracy while minimizing false positives and negatives.

2.4 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a supervised machine learning model that relies on dividing the surface of data points into subareas using decision lines known as hyperplanes. Typically, SVM works by finding the optimal hyperplane in a high-dimensional space that separates different classes of data points. The hyperplane is chosen to maximize the margin, which is the distance between the hyperplane and the closest data points from each class [28].

Several proposals have used SVM models to detect and identify cyberbullying in social media texts. For example, Purnamasari et al. [16] used SVM to distinguish between cyberbullying and non-cyberbullying instances in social media texts. Typically, offensive words were labeled with negative values, while non-offensive words were labeled with positive values. The labeled data was then used to train the SVM model to distinguish between both classes. The proposed model achieved an accuracy of 75% and a recall of 86.66%. Similar approaches were implemented by [29, 12]

to detect and identify cyberbullying on YouTube and Twitter, respectively. Both of these approaches achieved relatively high accuracy rates, with [29] reporting an accuracy of 75% and [12] reporting an accuracy of 85%.

Saranyanath et al. [8] used SVM and TF-IDF techniques together to detect and identify instances of cyberbullying. Typically, TF-IDF was used to extract additional features, such as the number of consecutive offensive words, and the SVM model was then trained on those features to identify cyberbullying instances. The proposed model achieved an accuracy rate of 86%. A similar approach was implemented by Bhagya et al. [30] to detect cyberbullying in the comments section of Wikipedia, achieving a high accuracy of 92%.

SVM is implemented in this research since it can effectively distinguish between cyberbullying and noncyberbullying texts, even in the presence of noisy or overlapping data.

2.5 Long Short Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a specialized type of recurrent neural network (RNN) that is designed to efficiently store and recall long-term dependencies in sequential data. As a result, it is a powerful deep learning model that is well-suited for a range of complex applications, including language modeling, speech recognition, and time-series prediction [31, 32].

Researchers have widely used LSTM for detecting cyberbullying in social media texts. Dessi et al. [33] used LSTM to develop an automated approach for identifying toxicity and unethical behavior in social media. Their research achieved an impressive accuracy of 91%. Dass et al. [34] utilized LSTM to identify cyberbullying in texts

from Twitter and Kaggle. Their research yielded a 75% accuracy rate for detecting cyberbullying texts.

Agarwal et al. [35] employed three deep learning models, including CNN, LSTM, and BiLSTM, to detect instances of cyberbullying related to racism, sexism, and personal attack across three different platforms: Formspring, Twitter, and Wikipedia. Their research found that LSTM achieved the highest accuracy rate of 91% in detecting instances of racism and personal attacks, and an accuracy rate of 84% in detecting instances of sexism. Al Hasan et al. [36] employed four deep learning models, including LSTM, CNN+LSTM, GRU, and CNN+GRU, to detect instances of hate speech in Arabic tweets. Their research found that the LSTM model achieved a recall of 74% and an F1-score of 72%. Balakrishna et al. [37] employed multiple deep learning models, such as LSTM, BiLSTM, GRU, BiGRU, RMN, and BiRMN, to identify instances of cyberbullying in tweets. Of all the models used, LSTM achieved the highest accuracy of 90% in predicting instances of cyberbullying.

In summary, LSTM is an efficient technique for identifying cyberbullying in social media texts, thanks to its ability to track and retain important information in long sequences of text [31]. Specifically, LSTM can successfully model the relationships between words and phrases to detect patterns indicative of cyberbullying. Moreover, LSTM demonstrates efficient capabilities for learning complex patterns from input data, such as the sentiment behind a cyberbullying text. Overall, the use of LSTM can significantly aid in the detection of cyberbullying texts and contribute to creating a safer online environment. Because of the above reasons, LSTM is utilized for predicting cyberbullying texts in this research.

2.6 Summary

The literature review reveals three limitations of current proposals for identifying cyberbullying. Firstly, these approaches are primarily limited to a binary classification of text as cyberbullying or non-cyberbullying. They are unable to determine specific types of cyberbullying or the underlying reasons why certain individuals may be targeted based on characteristics such as gender, religion, age, or ethnicity. This is due to the use of supervised learning algorithms that require labeled data for training. Mainly, if the labeled data only contains information about whether a text is cyberbullying or not, then the model will only be able to make binary classifications. Secondly, these approaches rely on black-box machine learning models that do not provide explanations for their predictions. This lack of transparency makes it difficult for end-users to understand and interpret the reports generated by these models. Finally, the classification accuracy of these approaches is sometimes unsatisfactory.

To address these limitations, this thesis proposes using an XAI technology called Local Interpretable Model-agnostic Explanations (LIME) in conjunction with machine learning models. LIME provides justifications and explanations that facilitate understanding of the decision-making process. The machine learning models are fed with the LIME explanations to detect and identify different types of cyberbullying. To the best of our knowledge, this is the first work that leverages XAI capabilities to detect and identify various types of cyberbullying in a social media text.

Chapter 3

Cyberbullying Text Detection Methodology

This chapter describes the proposed approach to detecting instances of cyberbullying in texts published on social media platforms, such as messages and comments. This approach is built into three layers: Text Pre-processing, Text Training, and Decision Explanation.

3.1 Dataset

The study relies on a publicly accessible dataset published on Kaggle [38]. The dataset comprises a total of 47,000 plain-text tweets and is structured into two columns: the first column, labeled "tweet text" contains plain-text tweets from diverse Twitter accounts, while the second column, labeled "cyberbullying type" distinguishes between cyberbullying and noncyberbullying tweets.

3.2 Text Pre-processing Layer

The Text Pre-processing Layer in the cyberbullying detection system uses three Natural Language Processing (NLP) techniques: text cleaning, sentiment analysis, and

Term Frequency-Inverse Document Frequency (TF-IDF). Text cleaning is used to remove special characters, punctuation, and stopwords from all the texts in the dataset. Sentiment analysis is used to evaluate the emotional tone of a text, while TF-IDF is used to identify the importance of words in a text. These NLP techniques are implemented in our research to produce training data, which the machine learning models use to predict cyberbullying texts. Figure 3.1 illustrates a figure of the Text Pre-processing Layer that includes the objective of each step.

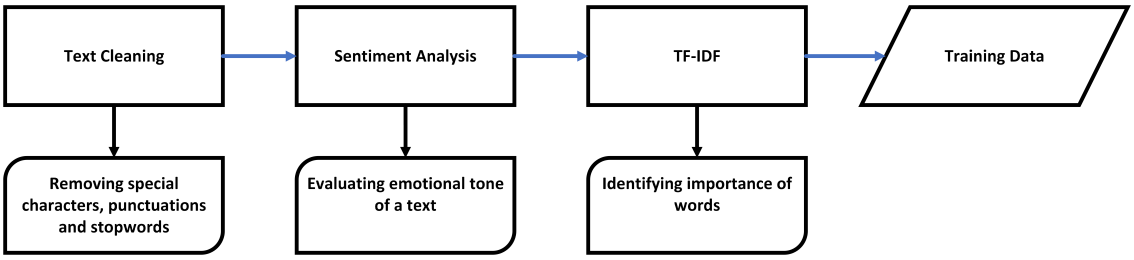


Figure 3.1: Text Preprocessing Layer

3.2.1 Text Cleaning Stage

The plain-text tweets in the dataset contain noises such as punctuation and special characters that negatively affect the efficiency of machine learning in terms of accuracy [39]. Specifically, text noises hinder machine learning models from analyzing the data smoothly and recognizing human writing patterns [8]. This raises the need to add a stage to clean the texts and filter out the noises.

In this stage, we utilize the package of Natural Language Toolkit (NLTK) [40] from NLP technology to clean the texts as follows:

1. **Filtering out the noise**: NLTK removes punctuation and special characters, for example, #, !, ?, from each text in the dataset.

2. **Removing stop words**: NLTK removes the stopwords, such as 'is', 'are', and 'and', from each text in the dataset. For example, the text "I hate you and you are worthless" becomes "I hate you you worthless" after removing stopwords. This step enables the machine learning models to focus highly on the offensive keywords associated with cyberbullying [39].

3. **Converting letters into lowercase**: NLTK converts all the texts in the dataset into lowercase letters to eliminate the effect of the case sensitivity on the outputs of the machine learning models.

4. **Tokenizing texts**: NLTK breaks each text in the dataset into separate words. For example, the text "I hate you you worthless" is broken into 'I', 'hate', 'you', 'you' and 'worthless'.

5. **Converting words into its roots**: NLTK converts all the words in each text in the dataset to their root form [41]. For example, the word 'bullying' becomes 'bully'. This step reduces the number of input words for the machine learning model to be processed, which consequently reduces efficiently time and memory space complexity.

3.2.2 Sentiment Analysis

In this research, the emotional tone of each text (or record) in the dataset is evaluated using a sentiment analysis tool from NLP technology. This tool assigns a sentiment score to each word in a text by consulting a lexicon-based dictionary and its corresponding score for the given word [42, 43]. For example, the word 'stupid' has a sentiment score of -1 in the lexicon-based dictionary. The overall sentiment score for

a given text (a record) is calculated according to Equation 3.1 as follows:

$$\delta_\tau = \frac{\delta_{\omega 1} + \delta_{\omega 2} + \ldots + \delta_{\omega i} + \ldots + \delta_{\omega n}}{\eta_\tau} \tag{3.1}$$

where δ_τ is the total sentiment score for a text τ, $\delta_{\omega i}$ is the sentiment score for the i^{th} word in the text τ, and η_τ is the number of words in the text τ.

The sentiment score for a text (δ_τ) takes a value in the range of $[-1, 1]$, where a value of -1 indicates an extreme case of negative sentiment and a value of 1 indicates an extreme case of positive sentiment [43]. When the sentiment score δ_τ approaches 0, the text is deemed to express a neutral sentiment.

3.2.3 Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency-Inverse Document Frequency (TF-IDF), one of NLP's tools, is utilized to evaluate the weights of texts' words in the dataset [43, 15]. Specifically, Term Frequency (TF) score indicates the number of times a word appears in a text [15]. TF score of a word ω_i in a text τ_j is given in Equation 3.2 as follows:

$$TF(\omega_i, \tau_j) = \frac{f(\omega_i, \tau_j)}{\eta_{\tau_j}} \tag{3.2}$$

where $f(\omega_i, \tau_j)$ is the frequency of the word ω_i in the text τ_j and η_{τ_j} is the total number of words in the text τ_j.

After calculating the words' TF scores, we calculate Inverse Document Frequency (IDF) score for each word in the dataset. The IDF score of a word measure how common or rare a word is across all the texts in the dataset [15]. The IDF score of

the word ω_i is given by Equation 3.3 as follows:

$$IDF(\omega_i) = \log\left(\frac{N}{\Omega(\omega_i)}\right) \tag{3.3}$$

where N is the total number of texts in the dataset and $\Omega(\omega_i)$ is the number of texts that include the word ω_i.

To calculate the TF-IDF score of the word ω_i in the text τ_j, we multiply the TF and IDF scores [15], as specified in Equation 3.4.

$$TFIDF(\omega_i, \tau_j) = TF(\omega_i, \tau_j) * IDF(\omega_i, \tau_j) \tag{3.4}$$

To demonstrate the significance of the word ω_i in the text τ_j, we calculate its weight using Equation 3.5.

$$W(\omega_i, \tau_j) = \frac{TFIDF(\omega_i, \tau_j)}{TFIDF_{max}(\tau_j)} \tag{3.5}$$

where $W(\omega_i, \tau_j)$ is the weight of the word ω_i in the text τ_j, $TFIDF(\omega_i, \tau_j)$ is the TF-IDF score of the word ω_i in the text τ_j, and $TFIDF_{max}(\tau_j)$: The highest TF-IDF score among all words in the text τ_j.

3.2.4 Cyberbullying Prediction Score

Sentiment scores, TF-IDF scores, and weight scores of words are utilized to generate a cyberbullying prediction score for each text in the dataset. The approach combines the strengths of both sentiment analysis and TF-IDF, capturing both the importance of specific words and the sentiment behind those words.

The cyberbullying prediction score is used to predict whether a text τ_j includes cyberbullying content or not. This score is calculated using Equation 3.6, where weight scores are represented by $W(\omega_1, \tau_j)$, $W(\omega_2, \tau_j)$, ..., $W(\omega_n, \tau_j)$, sentiment scores are represented by δ_{ω_1}, δ_{ω_2}, ..., δ_{ω_n}, TF-IDF scores are represented by $TFIDF(\omega_1, \tau_j)$, $TFIDF(\omega_2, \tau_j)$, ..., $TFIDF(\omega_n, \tau_j)$, and $SentimentBias$ is a parameter that adjusts the text's overall positive or negative sentiment. If the predicted score is negative, it indicates that the text contains cyberbullying content.

$$\begin{aligned} PScore(\tau_j) =& W(\omega_1, \tau_j) * (\delta_{\omega_1} + TFIDF(\omega_1, \tau_j)) + W(\omega_2, \tau_j) * (\delta_{\omega_2} + TFIDF(\omega_2, \tau_j)) + \ldots + \\ & W(\omega_n, \tau_j) * (\delta_{\omega_n} + TFIDF(\omega_n, \tau_j)) + SentimentBias \end{aligned} \tag{3.6}$$

3.2.5 Numerical Example

To demonstrate the use of the equations associated with the text pre-processing layer, we use the text, "You loser clearly I hate you" as a motivating example. Furthermore, in the training dataset of 47,000 texts (records), the words 'loser' and 'hate' appear in 105 and 156 records, respectively. As easily noted, the input text includes six words. The word 'You' appears twice in the text, while the remaining words appear only once.

The sentiment scores for all words in the text are calculated and listed in Table 3.1.

The TF-IDF scores for all the words in the input text are calculated using Equations 3.2, 3.3, and 3.4, and then listed in Table 3.2. As indicated in the table, the word "loser" has the highest TF-IDF score, resulting in a $TFIDF_{max}$ value of 0.44.

The weights for all words in the input text are calculated using Equation 3.5,

Table 3.1: Sentiment Scores

Word	Sentiment Score	Sentiment Label
You	0	Neutral
loser	-1	Negative
clearly	1	Positive
I	0	Neutral
hate	-1	Negative

Table 3.2: TF-IDF Scores

Word	TF	IDF	TF*IDF
You	2/6	log(47000/15932)	0.15
loser	1/6	log(47000/105)	0.44
clearly	1/6	log(47000/5062)	0.16
I	1/6	log(47000/20000)	0.061
hate	1/6	log(47000/156)	0.41

and then listed in Table 3.3. The sentiment bias for the input text is calculated by Equation 3.7.

Table 3.3: Weight Scores

Word	TFIDF	TFIDFmax	Weight
You	0.15	0.44	0.34
loser	0.44	0.44	1
clearly	0.16	0.44	0.35
I	0.061	0.44	0.13
hate	0.41	0.44	0.93

$$\begin{aligned} \delta_\tau &= \frac{\delta_{\omega 1} + \delta_{\omega 2} + \ldots + \delta_{\omega i} + \ldots + \delta_{\omega n}}{\eta_\tau} \\ &= \frac{0 + (-1) + 1 + 0 + (-1)}{6} \\ &= -0.16 \end{aligned} \tag{3.7}$$

Finally, the predicted value, to determine whether the text is a cyberbullying text or noncyberbullying, is calculated using Equation 3.6 as follows.

$$\begin{aligned} PScore(\tau_j) =& W(\omega_1, \tau_j) * (\delta_{\omega 1} + TFIDF(\omega_1, \tau_j)) + W(\omega_2, \tau_j) * (\delta_{\omega 2} + TFIDF(\omega_2, \tau_j)) + \ldots + \\ & W(\omega_n, \tau_j) * (\delta_{\omega n} + TFIDF(\omega_n, \tau_j)) + SentimentBias \\ & = 0.34 * (0 + 0.15) + 1 * (-1 + 0.44) + 0.35 * (1 + 0.16) + 0.13 * (0 + 0.061) + \\ & 0.93 * (-1 + 0.41) + (-0.16) \\ & = -0.80 \end{aligned} \tag{3.8}$$

The value of the prediction score is -0.80, which indicates that the input text is considered cyberbullying.

3.3 Text Training Layer

Four machine learning models, Random Forest, Gradient Boosting, Support Vector Machine (SVM), and Long Short Term Memory (LSTM), are used to predict cyberbullying texts based on the TF-IDF and sentiment scores of each text in the dataset. We selected the above models because of their ability to handle noisy and large amounts of data and the complex relationships among input features, such as the relationship between sentiment and TF-IDF scores [26, 44]. As a result, they are more reliable and accurate for detecting cyberbullying text. The key idea is to represent the text in a numerical form that captures each text's sentiment and TF-IDF scores. Then machine learning models use this information to predict whether a text is cyberbullying or noncyberbullying [26]. Figure 3.2 illustrates the Text Training Layer of our cyberbullying detection system.

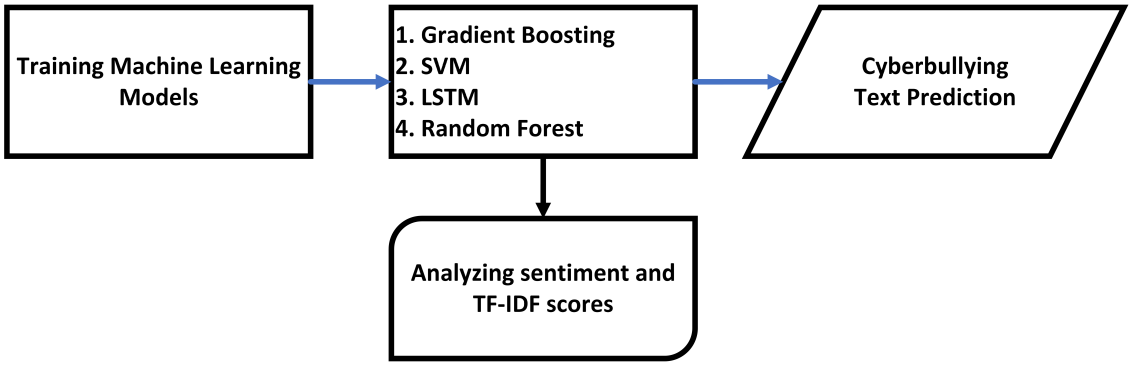


Figure 3.2: Text Training Layer

3.3.1 Gradient Boosting

The Gradient Boosting model processes and classifies the text as cyberbullying and non-cyberbullying according to the following steps.

1. **Input**: The sentiment and TF-IDF scores of each text are used as input for training the Gradient Boosting model [25].

2. **Initializing the training model**: In this step, the base predicted and actual values for each text (a record) are assigned in the training dataset. The base predicted value ($\hat{y}_{base}$) is calculated using Equation 3.6. Then, an actual value (-1 or 1) is assigned to represent whether the text is cyberbullying or non-cyberbullying, respectively. For example, the base predicted value ($\hat{y}_{base}$) for the input text "You loser clearly I hate you" is -0.80, i.e., $\hat{y}_{base} = -0.80$, while the actual value y is -1, i.e., $y = -1$.

3. **Calculating residual errors**: The residual error (E) for each text is calculated by subtracting the predicted value $\hat{y}$ from the actual value y, as given by Equation 3.9.

$$E = y - \hat{y} \tag{3.9}$$

For example, the residual errors for the input text "You loser clearly I hate you" is calculated as follows:

$$E = -1 - (-0.80) = -0.20 \tag{3.10}$$

4. **Training the model on residual errors**: The model is trained on the residual errors to generate a new predicted value $\hat{y}_{new}$ closer to the actual value, as given by Equation 3.11. Here, α represents the learning rate, which is often assigned as 0.1.

$$\hat{y}_{new} = \hat{y}_{base} + \alpha * E \tag{3.11}$$

For example, the new predicated value for the input text "You loser clearly I hate you" is calculated as follows:

$$\hat{y}_{new} = -0.80 + (0.1 * -0.20) = -0.82 \tag{3.12}$$

5. **Minimizing the residual errors**: In this step, Steps 3 and 4 are repeated until the minimum residual error is reached.

3.3.2 Support Vector Machine (SVM)

A binary training set is created for each text class in the dataset. In the binary training set, texts labeled as cyberbullying have a value of 1, while those labeled as non-cyberbullying have a value of -1. SVM creates a hyperplane with a decision boundary using the binary training set to separate our training data with a maximum margin [45]. Cyberbullying texts with a value of 1 are on one side of the

hyperplane, while non-cyberbullying texts with a value of -1 are on the other side of the hyperplane.

3.3.3 Long Short Term Memory (LSTM)

Long Short Term Memory (LSTM) is implemented to predict cyberbullying texts with the help of the LSTM algorithm. Algorithm 1 presents the pseudocode of the LSTM algorithm to predict cyberbullying texts [31].

Algorithm 1 LSTM Model for Cyberbullying Text Prediction

1: **procedure** LSTM MODEL
2: Preprocess input text
3: Compute Sentiment and TF-IDF scores
4: **Embedding Layer:** Convert sentiment and TF-IDF scores into representation vectors
5: **LSTM Layer:** Process input vectors using the following equation: $L\_O = \sum(x\_t * W) + SentimentBias$
6: **Dense Layer:** Apply Sigmoid activation function to the output of the previous step as follows: $y = \sigma(\delta(units = 1)(L\_O))$
7: **Loss Function:** Calculate Binary Cross Entropy Loss Function according to the following equation: $B\left(a, p\right) = -\left(a^{*} \log\left(p\right) + \left(1 - a\right)^{*} \log\left(1 - p\right)\right)$
8: **Optimizer:** Use Adam Optimizer to update network weights
9: Train the LSTM model
10: **end procedure**

The LSTM model uses the following steps to predict cyberbullying texts:

1. **Embedding Layer**: In this layer, the sentiment and TF-IDF scores of each text are converted to a representation vector, namely by embedding [32]. The LSTM model uses those embedding to understand the relationships between sentiment and TF-IDF scores for predicting a cyberbullying text. For example, TF-IDF and sentiment scores of the text "You loser clearly I hate you" are combined and converted into a representation vector as follows: [0.15, 0.44, 0.16, 0.061,

0.41, 0, -1, 1, 0, -1]. Thereafter, the LSTM model passes the embedding to the next layer (LSTM layer) to predict whether the text is cyberbullying or noncyberbullying.

2. **LSTM Layer**: The LSTM layer processes representation vectors to predict cyberbullying text [32] according to Equation 3.13.

$$L\_O = \sum(x\_t * W) + SentimentBias \tag{3.13}$$

where L_O is the output which is either cyberbullying or noncyberbullying, x_t is the input representation vector, W is the weight of each word in the input text, and $SentimentBias$ is the overall sentiment of the input text.

For example, to produce the output for the text "You loser clearly I hate you" the LSTM layer uses the above equation as follows:

$$\begin{aligned} L\_O = &\sum(x\_t * W) + SentimentBias \\ &= [0.15, 0.44, 0.16, 0.061, 0.41, 0, -1, 1, 0, -1] \\ &* [0.34, 1, 0.35, 0.13, 0.93, 0, 0, 0, 0, 0] + (-0.16) \\ &= [0.051 + 0.44 + 0.056 + 0.00793 + 0.3813 + 0 + 0 + 0 + 0 + 0] + (-0.16) \\ &= 0.94 - 0.16 \\ &= 0.78 \end{aligned} \tag{3.14}$$

where the input representation vector of the text is [0.15, 0.44, 0.16, 0.061, 0.41, 0, -1, 1, 0, -1], the weight scores for the words in the text are (0.34, 1, 0.35, 0.13, 0.93), and the sentiment bias is -0.16. To perform a matrix multiplication

between the input vector and the weight scores, the weight scores vector becomes [0.34, 1, 0.35, 0.13, 0.93, 0, 0, 0, 0, 0]. The point behind this step is to match the length of the weight scores vector with the length of the input vector. The LSTM layer adds the sentiment bias (−0.16) to produce the output.

3. **Dense Layer**: The Dense Layer, namely by Fully connected layer, applies the Sigmoid activation function to calculate the probability of a text being a Cyberbullying [46, 32]. The Sigmoid activation function is given by Equation 3.15 as follows.

$$y = \sigma(\delta(units = 1)(L\_O)) \tag{3.15}$$

Where L_O is the output vector of our LSTM model, δ is the dense layer, $units = 1$ is the number of output units with a value 1 in the Dense layer, and σ is the Sigmoid activation function.

For example, the probability that the text "You loser clearly I hate you" is cyberbullying is calculated using the Sigmoid function as follows:

$$\begin{aligned} y =& \sigma(\delta(units = 1)(L\_O)) \\ &= \sigma(0.78) \\ &= 0.68 \end{aligned} \tag{3.16}$$

Here, the Sigmoid function produces a result of 0.68, which is greater than the threshold value of 0.5. Therefore, LSTM predicts the text "You loser clearly I hate" as cyberbullying.

4. **Loss Function**: The loss function, Binary cross-entropy, is applied to measure

the accuracy of the LSTM model [47]. Mainly, Binary cross-entropy compares the predicted value generated by the LSTM layer with respect to the actual value in the training dataset [47]. The Binary cross-entropy function is given by Equation 3.17 as follows:

$$B\left(a,p\right) = -\left(a^{*}\log\left(p\right) + \left(1-a\right)^{*}\log\left(1-p\right)\right) \tag{3.17}$$

where B represents the Binary cross-entropy loss function, a is the actual value (label) for the input text, and p is the predicted value (label) for the input text.

For example, the actual value of the text "You loser clearly I hate you" is 1 in the training dataset, while the value predicted by the LSTM model is 0.68. Therefore, the Binary cross-entropy loss function value for the input text is given by Equation 3.18. The loss value for the input text is 0.167, which refers to the high efficiency of the LSTM model as the loss value is small.

$$\begin{aligned} B(1,0.68) &= -\left(1^{\star}\log(0.68) + (1-1)^{*}\log(1-0.68)\right) \\ &= -\left(1^{*}0.167 + 0^{*}0.167\right) \\ &= 0.167 \end{aligned} \tag{3.18}$$

5. **Adam Optimizer**: Adam Optimizer is an optimization algorithm that is used to update network weights during the training process [47]. It plays a crucial role in increasing the accuracy of the LSTM model by minimizing the value of the loss function [47]. It also speeds up the training process, which decreases the computational time.

3.3.4 Random Forest Classifier (RFC)

The following steps are used to build a Random Forest Classifier to predict cyberbullying texts from the above example:

1. **Input:** First, texts are preprocessed in the dataset to remove punctuations, special characters, and stopwords. Then, the sentiment and TF-IDF scores are calculated for each text. A prediction score of each text in the dataset is calculated using Equation 3.6 which combines both sentiment and TF-IDF scores. Finally, prediction scores are used as inputs to train a Random Forest model.

For example, if the preprocessed input text is "You loser clearly I hate you" and the prediction score for the above text is -0.8, the random forest model would predict whether the above text is cyberbullying or not based on the prediction score.

2. **Decision Trees:** 500 decision trees are used to develop the random forest model for predicting cyberbullying texts so that the trees can improve the overall performance and robustness of the model.

3. **Bootstrap Aggregation:** Random Forest uses bootstrap aggregation or bagging to create multiple training sets from our dataset using different subsets of training data and features [20]. Each training set is built by randomly selecting prediction scores and labels from the dataset. The concept behind bootstrap aggregation is that we can improve the model's performance and reduce model variance by building additional training sets from the original data [20]. For example, Random Forest creates the following bootstrap samples from our dataset by randomly selecting prediction scores and labels from the texts in our dataset:

4. **Gini Impurity:** Following the creation of bootstrap samples, Random Forest creates multiple decision trees on each sample based on the random selection of the

Table 3.4: Bootstrap Sample

Text	Score	Label
You loser clearly I hate you	-0.76	Cyberbullying
You are a beautiful person	1.19	Noncyberbullying
I am an unbiased person	0.35	Noncyberbullying
I hate your guts	-0.69	Cyberbullying
Kate is a stupid person	-0.45	Cyberbullying

prediction scores [20]. For example, in our bootstrap samples, we want to build a decision tree to classify the texts as cyberbullying or noncyberbullying based on prediction scores. Random Forest randomly selects the prediction scores for each node and then splits each node using a technique called Gini Impurity [48]. Gini Impurity scores range from 0 to 0.5. The lowest Gini Impurity score is used to select the best split for splitting a node. If a node receives a Gini Impurity score of 0, it is considered the leaf node and used to make a prediction [48]. The following equation is used to calculate Gini Impurity to select leaf nodes in the RFC model in the dataset:

$$GiniImpurity = 1 - ({P_{cyberbullying}}^2 + {P_{noncyberbullying}}^2) \tag{3.19}$$

where $Pcyberbullying$ represents the proportion of cyberbullying class and $Pnoncyberbullying$ represents the proportion of noncyberbullying class.

5. **Decision Tree Predictions:** A Random Forest Classifier uses bootstrap aggregation to train several decision trees on various subsets of the data. Each decision tree develops its own patterns through independent predictions [20]. As a result, individual decision trees may produce different predictions for the same input text.

Random Forest uses Gini Impurity to choose the conditions for the splits during the construction of the decision tree [20]. The split conditions are based on the

features and their values. For example, in the bootstrap sample, the input text is "You loser clearly I hate you" and the prediction score is -0.76. The score is generated using Equation 3.6, which combines the values of the features, such as sentiment and TF-IDF scores. Each decision tree uses different feature values and split conditions to make predictions. For the text "You loser clearly I hate you", decision tree 1 classifies it as cyberbullying due to the negative sentiment score of the text, while decision tree 2 classifies it as cyberbullying due to the high TF-IDF scores of the offensive words, such as loser and hate.

6. **Final Prediction:** Random Forest makes the final prediction of each text in the dataset by combining the predictions of all the decision trees and by taking the majority vote of each class across all the trees.

For example, To predict whether the text."You loser clearly I hate you" from our example texts, the trained decision tree models make the following predictions:

Decision tree 1: cyberbullying

Decision tree 2: non-cyberbullying

Decision tree 3: cyberbullying

The majority vote of the predictions would be 2 out of 3, or 67%, in favor of cyberbullying. Therefore, the final prediction of the random forest model for this text would be cyberbullying. Random Forest keeps calculating the majority vote until it makes the final predictions for all the texts in the dataset.

3.4 Decision Explanation Layer

In Decision Explanation Layer, the Local Interpretable Model-agnostic Explanations (LIME) algorithm from XAI technology is utilized to explain and justify the decisions

taken by the four machine learning models in the second layer. First, LIME generates explanations of cyberbullying predictions, including the weights for influential words in a cyberbullying text. Next, these influential words are utilized to re-label the cyberbullying texts in the dataset based on the following classifications: gender, religion, age, and ethnicity. The machine learning models are then re-trained to predict multiclass cyberbullying texts. Finally, LIME is employed to generate explanations of multiclass cyberbullying texts to show the important multiclass cyberbullying words and compare the performances of machine learning models. Figure 3.3 graphically illustrates the Decision Explanation Layer.

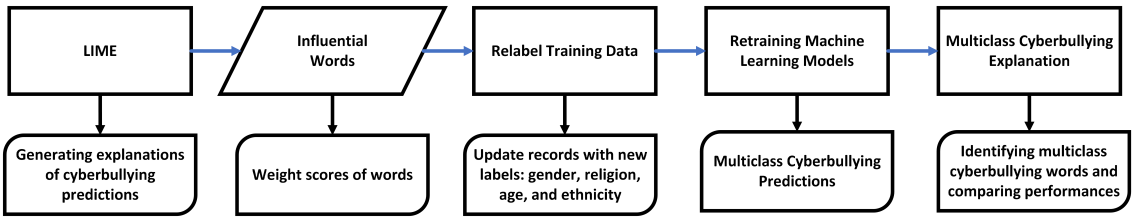


Figure 3.3: Decision Explanation Layer

3.4.1 LIME

A common algorithm in Explainable Artificial Intelligence (XAI) is LIME, which aims to provide explanations that humans understand for the predictions made by a machine learning model [49]. Algorithm 2 provides the pseudocode to explain cyberbullying predictions in this research [49, 50].

The LIME algorithm to explain the cyberbullying predictions works as follows.

1. **Producing perturbed texts**: When given a text containing cyberbullying content as input, LIME produces perturbed texts by randomly removing some words from the input text [49]. Taking the cyberbullying text "You loser clearly

Algorithm 2 Cyberbullying Prediction Explanation by LIME

Require: Selected Model's Predictions
Ensure: Important Features, Performance Reliability
1: Select Text to Explain: x
2: Create Perturbed Versions of Text: x'
3: Obtain Predictions from the Selected Model for Perturbed Text g
4: Calculate weights of each word in x':

$$Weight(w) = \left(1 * TFIDF(w,T) - 0.5 * \sum \left[TFIDF\left(w,T'\right)\right]\right)/k$$

5: Select important words based on weight scores
6: Train an Interpretable linear model
7: Obtain Predictions from Interpretable Model for Perturbed Text: m
8: Calculate Loss Function between g and m:

$$Ł(f,g,\pi_x) = \sum \left|f\left(x\_i\right) - g\left(x\_i\right)\right| \pi_x\left(x\_i\right)$$

9: Calculate Reliability Score:

$$R = 1 - L(f,g,\pi_x)$$

10: **if** R Score is High **then**
11: Selected Model's Performance is Reliable
12: **else**
13: Selected Model's Performance is Not Reliable
14: **end if**
15: Generate Report to feature important words and performance reliability

I hate you" as an example, LIME generates perturbed texts like "You loser I hate you" and "You clearly I hate you". In the first perturbed version, 'clearly' is randomly removed, while in the second, 'loser' is randomly removed. This step is repeated multiple times to create all possible perturbed texts.

2. **Calculating words' weight**: LIME calculates the weight of words by measuring the difference between their TF-IDF scores in perturbed texts and their

TF-IDF scores in the original text, as given in Equation 3.20.

$$Weight(w) = \Big(1 * TFIDF(w,T) - 0.5 * \sum [TFIDF\,(w,T')]\Big)\,/k \quad (3.20)$$

where $TFIDF(w,T)$ is the TF-IDF score of the word w in the original text T, $TFIDF\,(w,T')$ is the TF-IDF score of the word w in the Perturbed text T', and k is the total number of original and perturbed texts.

For example, LIME computes the weight of the word 'loser' in the text "You loser clearly I hate you" using Equation 3.21.

$$\begin{aligned} Weight(loser) &= \Big(1 * TFIDF(loser,T) - 0.5 * \sum [TFIDF\,(loser,T')]\Big)\,/k \\ &= (1 * 0.98 - 0.5 * (0.4 + 0))/3 \\ &= 0.26 \end{aligned} \quad (3.21)$$

where T represents the original text "You loser clearly I hate you", T' represents the perturbed texts "You loser I hate you" and "you clearly I hate you", $TFIDF(w,T)$ is the TF-IDF value of the word 'loser' in the original text which is 0.98, $TFIDF\,(w,T')$ are the TF-IDF values of the word 'loser' in the perturbed texts which are (0.4, 0), and k is the total number of original and perturbed texts which is 3.

3. **Generating explanations**: LIME selects the most influential words with the highest weights. For example, in the text "You loser clearly I hate you", the words 'loser' and 'hate' have the highest weights. These words significantly contribute to the decision of the machine learning model to classify the text

as cyberbullying. The words, along with their weights, are then included in a report as explanations for the decision made by the machine learning model.

4. **Training perturbed texts**: The perturbed texts serve as inputs for the four machine learning models in the second layer, which determine whether the perturbed texts constitute cyberbullying [49]. For instance, the perturbed text "You loser I hate you" is fed into the RFC model in the second layer and subsequently classified as cyberbullying. LIME creates new interpretable linear models and trains them on the perturbed texts. Afterward, LIME compares the accuracy score of the interpretable linear models with the accuracy score of the machine learning models in the second layer, as given by the loss function in Equation 3.22. The loss function Ł(f, g, π_x) measures the reliability of the decisions taken by the machine learning models in the second layer.

$$L(f, g, \pi_x) = \sum |f(x\_i) - g(x\_i)| \, \pi_x(x\_i) \tag{3.22}$$

where $f(x\_i)$ is the Original Model's prediction accuracy score, $g(x\_i)$ is the Interpretable Model's prediction accuracy score, and $\pi_x(x\_i)$ is the similarity between all the words in the Original model and the interpretable model.

For example, consider the perturbed texts "You loser I hate you" and "You completely I hate you" used as inputs for an RFC (Random Forest Classifier) model. The RFC model predicts the following probabilities for cyberbullying for both texts:

- "You loser I hate you" = 0.8 (cyberbullying)
- "You completely I hate you" = 0.5 (cyberbullying)

The interpretable model, such as Linear Regression, provides the following predictions for cyberbullying probabilities for both texts:

- "You loser I hate you" = 0.75(cyberbullying)
- "You completely I hate you" = 0.4 (Cyberbullying)

Using the cosine similarity, we find out that the similarity value between the words in the original model and the interpretable model is 1 for the perturbed text "You loser I hate you" and 0.7 for the perturbed text "You completely I hate you". Therefore, the loss function between the original model and the interpretable model is:

$$\begin{aligned} Ł(f, g, \pi_x) &= \sum |f\left(x\_i\right) - g\left(x\_i\right)| \, \pi_x\left(x\_i\right) \\ &= |0.8 - 0.75| * 1 + |0.5 - 0.4| * 0.7 \\ &= 0.12 \end{aligned} \tag{3.23}$$

The difference in the loss function between the original and interpretable models is 0.12, which indicates a low value. Now, LIME calculates the reliability score of the model using the following equation:

$$\begin{aligned} R =& 1 - Ł(f, g, \pi_x) \\ &= 1 - 0.12 \\ &= 0.88 \end{aligned} \tag{3.24}$$

Where R represents the reliability score, as noted in the equation, RFC has a high-reliability score of 0.88, which indicates it is reliable at predicting the text

"You loser completely I hate you" as cyberbullying.

5. **Generating Report:** LIME generates a report that includes the reliability score, the prediction probability of a text being cyberbullying or noncyberbullying, and the most influential words with their weight scores.

3.4.2 Multiclass Cyberbullying Detection

In this stage, the explanations for binary classification are fed back into the training dataset to re-label the texts (records) based on one of the following categories: gender, religion, age, or ethnicity. The machine learning models are then re-trained on the updated dataset to perform multiclass classification. This step enables the machine learning models to identify the type of cyberbullying, such as gender-based, religion-based, age-based, or ethnicity-based. Algorithm 3 is developed in this research to detect multiclass cyberbullying texts.

The following steps are used in this research to detect multiclass cyberbullying texts:

1. **Multiclass Cyberbullying Word List:** We compile and utilize standard libraries, including Urban Dictionary, Wiktionary, Wikipedia, and GitHub [51, 52, 53, 54, 55], to retrieve and match the most common words associated with different types of cyberbullying, such as gender-based, religion-based, age-based, or ethnicity-based. Figure 3.4 displays a sample of the most common offensive words for each category of cyberbullying. Figure 3.4(a) represents gender-based cyberbullying words, 3.4(b) represents religion-based cyberbullying words, 3.4(c) represents age-based cyberbullying words, and 3.4(d) represents ethnicity-based cyberbullying words.

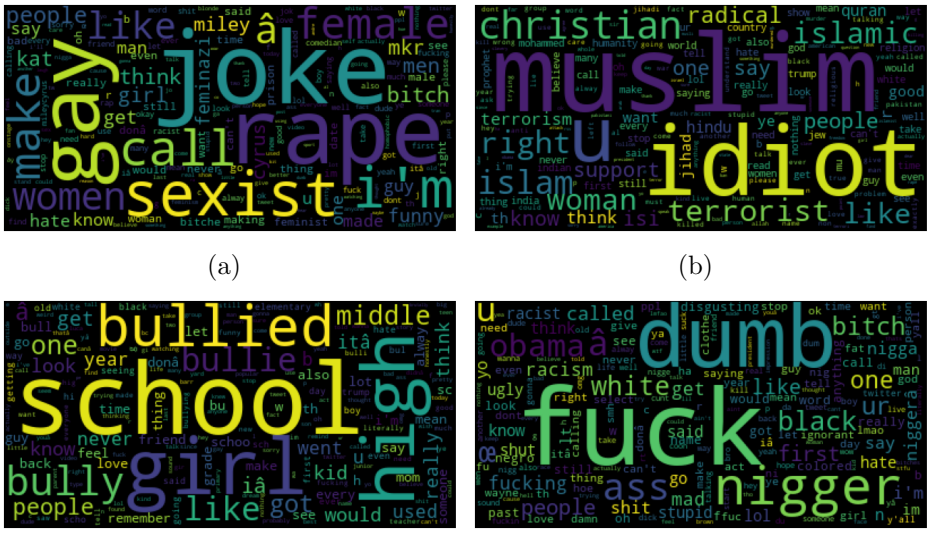
Algorithm 3 Multiclass Cyberbullying Detection using LIME

Require: X - Input texts, M - Machine learning models, L - LIME
Ensure: C - Cyberbullying classification

1: Compile multiclass word list using Urban Dictionary, Wiktionary, Wikipedia, and GitHub
2: **for** each *text* in X **do**
3: Obtain LIME explanation for *text*
4: Calculate weight scores for words in *text* using LIME explanation:

$$Weight(w) = \left(1 * TFIDF(w,T) - 0.5 * \sum [TFIDF\,(w,T')]\right)/k$$

5: Select the most important words based on weight scores
6: Match words with multiclass word list
7: Classify *text* based on matched words (gender, religion, age, or ethnicity)
8: Store classification results in C
9: **end for**
10: **for** each *model* in M **do**
11: Train *model* using X and C
12: Predict multiclass cyberbullying using *model*
13: Evaluate prediction accuracy
14: Obtain LIME explanation for multiclass prediction
15: **end for**



(a) (b)

(c) (d)

Figure 3.4: Multiclass Cyberbullying Words

2. **Generating Influential Cyberbullying words:** LIME technology is utilized to generate the most influential words, along with their corresponding weights, that affect the classification of a text as cyberbullying.

3. **Matching Words and Classifying Cyberbullying Texts:** The first three most influential words with the highest weight scores are selected and matched with the word list generated in step 1. As an example, let us consider the text "I am not sexist, I hate those women who accuse me of sexism" as an instance of cyberbullying. LIME identifies the words 'sexist', 'sexism', and 'women' as the most influential, with weight scores of 0.26, 0.23, and 0.29, respectively. By exploring the word lists generated in Step 1, these words are matched to the category of gender-based cyberbullying. Therefore, the text "I am not sexist, I hate those women who accuse me of sexism" is re-labeled as an instance of gender-based cyberbullying in the training dataset.

4. **Re-training Machine Learning Models:** The updated data is used to re-train the machine learning models for performing multiclass classification to identify the type of cyberbullying.

5. **Evaluating Performance:** LIME is used to explain and justify the multi-class classifications, which helps to evaluate the accuracy and reliability of the machine learning models.

3.5 Summary

In this chapter, we use the methods of NLP, machine learning, and XAI to develop a cyberbullying text detection system. Our system has three layers, Text Preprocessing Layer, Text Training Layer, and Decision Explanation Layer. In the Text Pre-processing Layer, our system first uses the text cleaning stage to remove noises from the texts in the dataset. Then it uses NLP tools, such as sentiment analysis and TF-IDF, to detect the importance of offensive words contributing to negative sentiments in cyberbullying texts. A numerical example is provided that shows how the sentiment and TF-IDF scores are calculated for each text and how we use the prediction score equation that combines both sentiment and TF-IDF scores to generate a prediction score for detecting a cyberbullying text.

In the Text Training Layer, our system implements four machine-learning models, Random Forest, Gradient Boosting, SVM, and LSTM, to predict cyberbullying texts based on sentiment and TF-IDF scores. We provide the necessary steps and examples to show how these machine learning models are used in the context of cyberbullying prediction.

In the Decision Explanation Layer, we use an XAI algorithm called LIME to generate explanations of cyberbullying predictions. The LIME explanations justify the cyberbullying predictions made by the machine learning models. We provide the steps and algorithm that show how LIME calculates the weight of each word in a text to show the importance of offensive words in a cyberbullying text and the reliability score of each prediction to determine whether the machine-learning models correctly predict cyberbullying words and texts.

Furthermore, we utilize the weight scores of words from Lime explanations to

classify a cyberbullying text based on gender, religion, age, and ethnicity. Machine learning models are then re-trained for multiclass cyberbullying prediction, and LIME is implemented to explain these predictions. This chapter outlines the steps and algorithm for detecting multiclass cyberbullying texts. The next chapter explores the experimental results obtained to evaluate our system's performance.

Chapter 4

Evaluation Results

This chapter discusses the comparative evaluation results of binary and multiclass cyberbullying texts to evaluate the performance of the proposed approach. The section is organized as follows: Section 4.1 discusses the evaluation metrics used to measure the performance of the machine learning models employed, Section 4.2 examines the performance of the machine learning models when performing binary classification for social media texts, Section 4.3 analyzes the explanations generated by the XAI algorithm, Local Interpretable Model-agnostic Explanations (LIME), to justify the decisions made by the machine learning models under the binary classification scenario, Section 4.4 evaluates the performance of the machine learning models for multiclass cyberbullying texts, Section 4.5 elaborates on the explanations produced by LIME, to explain the results of the multiclass cyberbullying text predictions, and finally Section 4.6 provides an overview of the external and internal validity of our proposed approach.

4.1 Evaluation Metrics

We evaluate the performance of the proposed approach according to the following four metrics: accuracy, recall, precision, and F- score [8].

1. Accuracy: The accuracy is measured as the proportion of the number of accurate predictions to the total number of predictions [8], as shown in the following equation.

$$Accuracy = \frac{T_p o + T_n e}{T_p o + F_p o + T_n e + F_n e} \tag{4.1}$$

where $T_p o$ is True Positive, $T_n e$ is True Negative, $F_p o$ is False Positive, and $F_n e$ is False Negative.

2. Precision: The precision is used to measure the ratio of true positives to the total predicted positives made by the machine learning models [26]. It is calculated as shown in the following equation.

$$Precision = \frac{T_p o}{T_p o + F_p o} \tag{4.2}$$

where $T_p o$ is True Positive and $F_p o$ is False Positive.

3. Recall: Recall is used to measure the ratio of true positives with respect to the total number of true positives and false negatives [56]. Recall is calculated as follows.

$$Recall = \frac{T_p o}{T_p o + F_n e} \tag{4.3}$$

where $T_p o$ is True Positive and $F_n e$ is False Negative.

4. F-Score: F-Score is used to integrate the precision and recall values into a single metric [56] as given in the following equation.

$$F - Score = \frac{2 \times P \times R}{P + R} \tag{4.4}$$

where P represents Precision and R represents Recall.

4.2 Binary Classification Performance

Table 4.1: Binary Cyberbullying Prediction Results

Model	Accuracy	Precision	Recall	F-Score
LSTM	92%	93%	92%	92%
SVM	90.67%	91%	91%	91%
Random Forest	86.84%	87%	85%	86%
Gradient Boosting	85.71%	86%	86%	86%

Table 4.1 presents the performance of the machine learning models used in the binary classification task. As shown in the table, the LSTM model achieves the highest accuracy, precision, recall, and F-Scores. Specifically, the accuracy score is 92%, and the precision score is 93%, while the recall and F-score are both 92%. The SVM and RFC models achieve the second and third highest performance, respectively. On the other hand, the Gradient Boosting model achieves the lowest performance, with an accuracy of 85.71%, and precision, recall, and F-Score results of 86%. Those results are graphically presented in Figure 4.1.

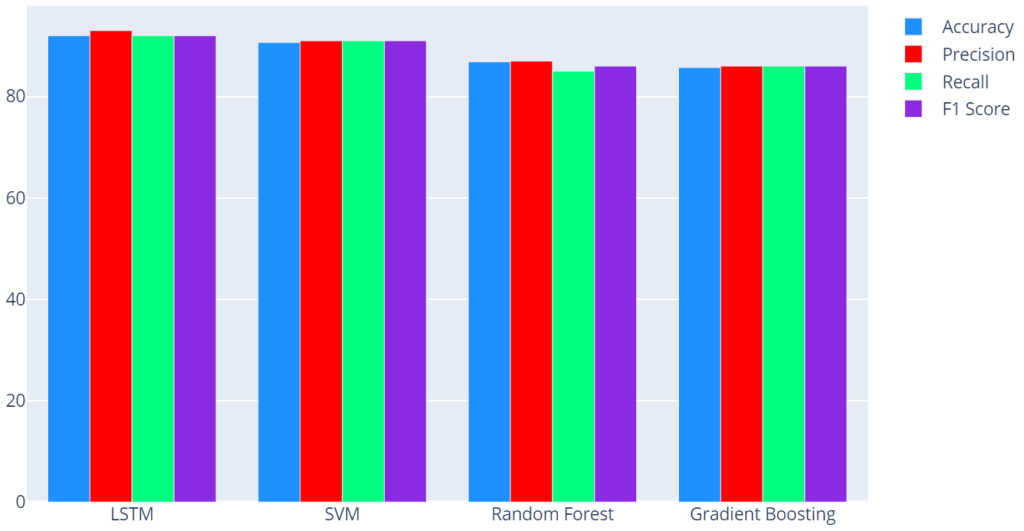


Figure 4.1: Performance Comparisons for Binary Cyberbullying

4.3 Binary Classification Explanation

In this section, we present the reliability score and the weights of the influential words that affect the decisions of the machine learning models. We also display the probability of potential outcomes, i.e., cyberbullying or non-cyberbullying. To compare the explanations generated by the XAI algorithm, LIME, for the machine learning model, we generated an explanation for the following tweet: "How about these idiots? You buffoon Indian Muslims. Everytime, you all are fooled. I just pity your low mind. #KashmiriLivesMatter". After preprocessing, the text has been transformed into "idiots buffoon Indian Muslims every time fool pity low mind KashmiriLivesMatter".

The explanations for RFC, Gradient Boosting, SVM, and LSTM are graphically presented in Figures 4.2, 4.3, 4.4, and 4.5 respectively. As illustrated in these figures,

the machine learning models have arrived at a consensus that the input text is indicative of cyberbullying with a high level of confidence, as indicated by their respective prediction probabilities. For instance, as shown in Figure 4.5, the LSTM model shows the highest level of confidence with a prediction probability of 100% that the input text is cyberbullying. In contrast, as shown in Figure 4.3, the Gradient Boosting model achieves the lowest confidence level among the models, predicting with a probability of only 62% that the input text is cyberbullying and likely classifying it as non-cyberbullying.

The machine learning models utilize LIME to exhibit similar justifications. For example, the words 'low', 'fool', and 'Muslims' are highlighted by all of the models as the most influential words affecting the classification of input text as cyberbullying. In Figure 4.2, for instance, the word 'low' has the highest weight at 0.24, followed by 'fool' at 0.14, and 'Muslims' at 0.09. These words have similar weights across all machine learning models, as illustrated in Figures 4.3, 4.4, and 4.5.

However, the contribution weights of the remaining words in the input text vary, affecting the decisions (classes) differently. For example, the offensive words 'pity', 'buffoon', and 'idiots' cause the Gradient Boosting model to miss-classify the input text as non-cyberbullying with a relatively high prediction probability of 0.38. This explains why, as shown in Figure 4.3, the Gradient Boosting model has the lowest reliability score (0.70) among the models, where the explanations do not match the decisions made. On the other hand, those offensive words contribute to the classification of the input text as cyberbullying by RFC, SVM, and LSTM. The RFC model obtains the highest reliability (0.98) due to the highly reasonable match between the explanations and the decisions taken. For instance, in Figure 4.2, all offensive

words contribute to the classification of the text as cyberbullying, while neutral words contribute to the classification of the text as non-cyberbullying.

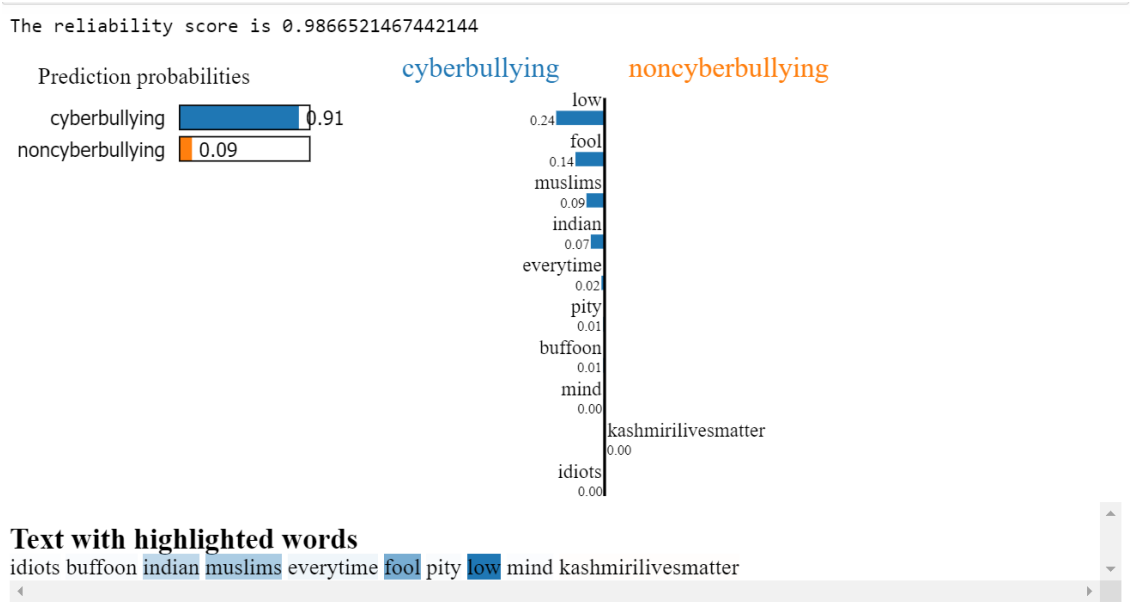


Figure 4.2: LIME Explanations for RFC Decisions

4.4 Multiclass Classification Performance

Table 4.2: Multiclass Cyberbullying Prediction Results

Model	Accuracy	Precision	Recall	F-Score
LSTM	97.5%	98%	99%	97%
SVM	95.25%	96%	96%	96%
Random Forest	95%	94%	96%	96%
Gradient Boosting	94.75%	95%	95%	95%

The results of the multiclass classification for cyberbullying texts are shown in Table 4.2. As can be seen from the table, all models exhibit better performance in terms of accuracy, precision, recall, and F-score compared to the binary classification

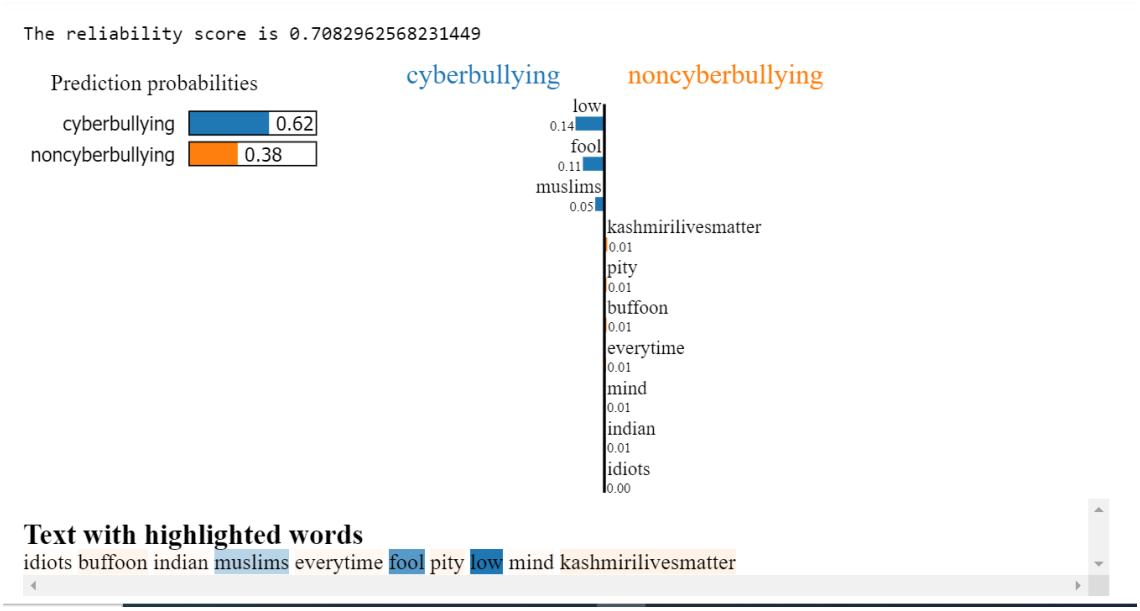


Figure 4.3: LIME Explanations for Gradient Boosting Decisions

case (see Table 4.1). For example, the Random Forest model shows an improvement in accuracy from 86.85% to 95.25%, precision from 87% to 94%, recall from 85% to 96%, and F-score from 86% to 96%. Similarly, the Gradient Boosting model shows a significant improvement in accuracy from 85.71% to 94.75%, precision from 86% to 95%, recall from 86% to 95%, and F-score from 86% to 95%. Those observations are graphically represented in Figures 4.6, 4.7, 4.8, 4.9 and 4.10.

4.5 Multiclass Classification Explanation

In this section, we demonstrate the explanations generated by LIME when the machine learning models perform multiclass classification on text data that includes instances of cyberbullying. As in the previous sections, we will compare the explanations in terms of their reliability score, prediction probability, and the weights of the most influential words that affect the machine learning decisions.

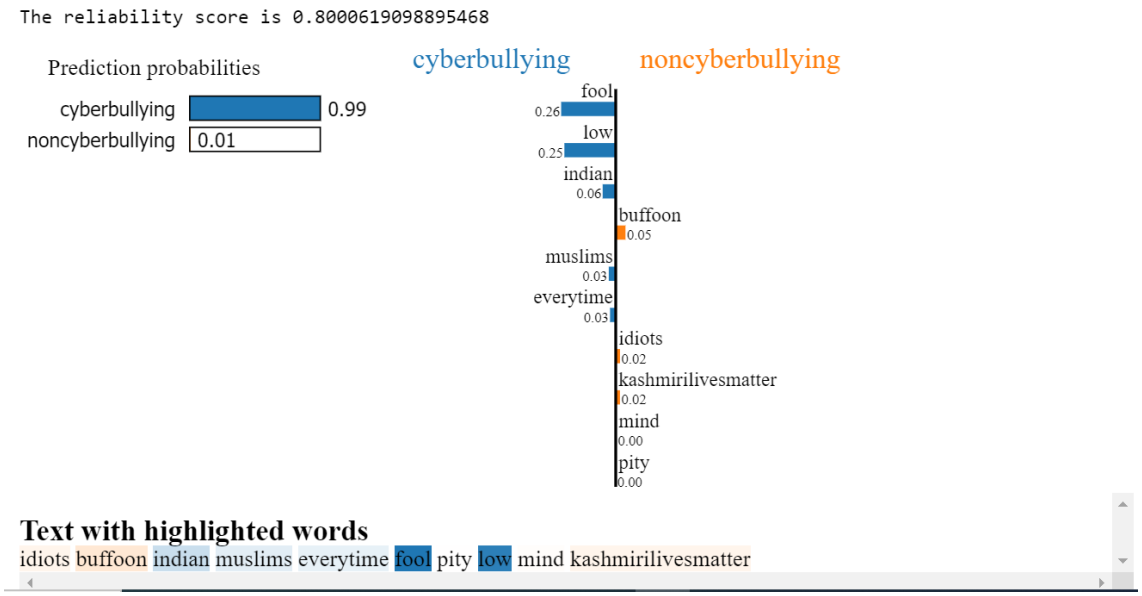


Figure 4.4: LIME Explanations for SVM Decisions

4.5.1 Gender-Based Cyberbullying Explanation

In this section, we aim to compare the explanations behind the decisions taken by the machine learning models when they are used to detect gender-based cyberbullying. To accomplish this, we generate the explanations for the following tweet: "Females and guys. @AwkwardEP I'm not sexist but a lot of females lack true logic sometimes. A lot of guys are just dumb though". Once this text is processed, it is transformed into "females guy sexist lot females lack true logic sometimes lot guy dumb though".

The explanations for RFC, Gradient Boosting, SVM, and LSTM are graphically presented in Figures 4.11, 4.12, 4.13, and 4.14 respectively. As illustrated in these figures, the machine learning models have arrived at a consensus that the input text is indicative of gender-based cyberbullying with different levels of confidence, as indicated by their respective prediction probabilities. For instance, as shown in Figure 4.13, the SVM model shows the highest level of confidence with a prediction

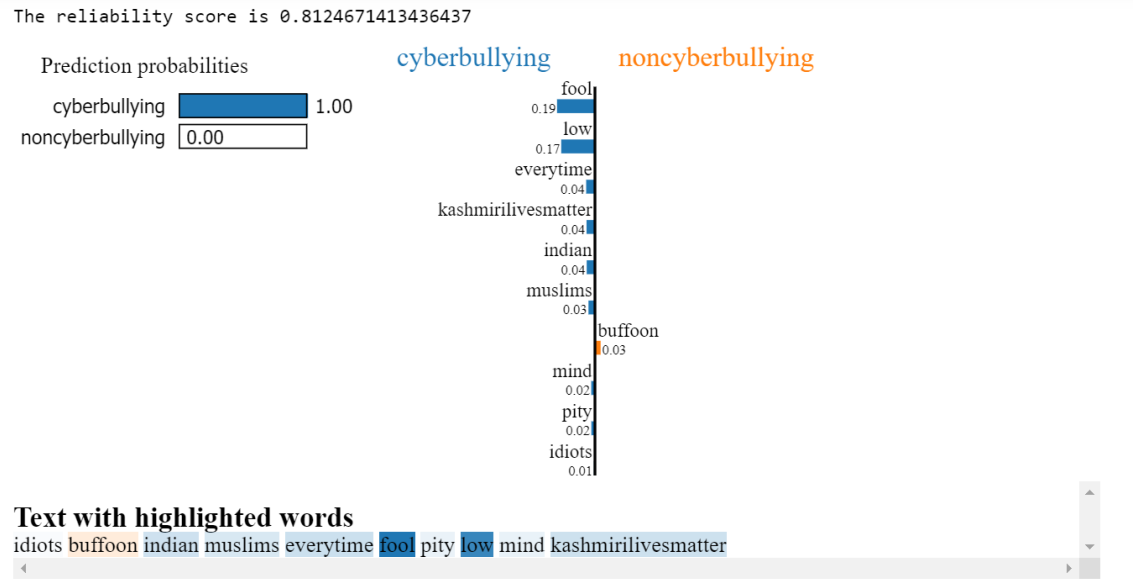


Figure 4.5: LIME Explanations for LSTM Decisions

probability of 97% that the input text is gender-based cyberbullying. In contrast, as shown in Figure 4.3, the Gradient Boosting model achieves the lowest confidence level among the models, predicting with a probability of only 54% that the input text is gender-based cyberbullying.

Using LIME, these machine learning models demonstrate similar justifications for explaining the predictions. For example, the words 'females', 'guy', and 'sexist' are highlighted by RFC, SVM, and LSTM models as influential words affecting the classification of input text as gender-based cyberbullying. In Figure 4.14, for instance, the word 'females' has a weight of 0.11, followed by 'guy' at 0.08, and 'sexist' at 0.04. These words have similar weights across those machine learning models (RFC, SVM, LSTM), as illustrated in Figures 4.11, 4.13, and 4.14.

On the other hand, the words 'females', 'guy', and 'sexist' cause the Gradient Boosting model to misclassify the input text as religion-based cyberbullying with a

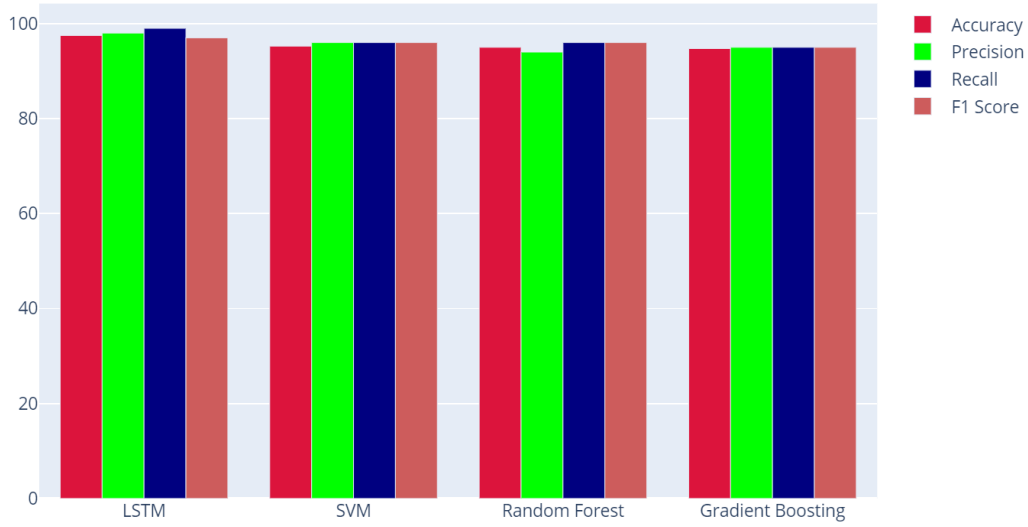


Figure 4.6: Performance Comparisons for Multiclass Cyberbullying

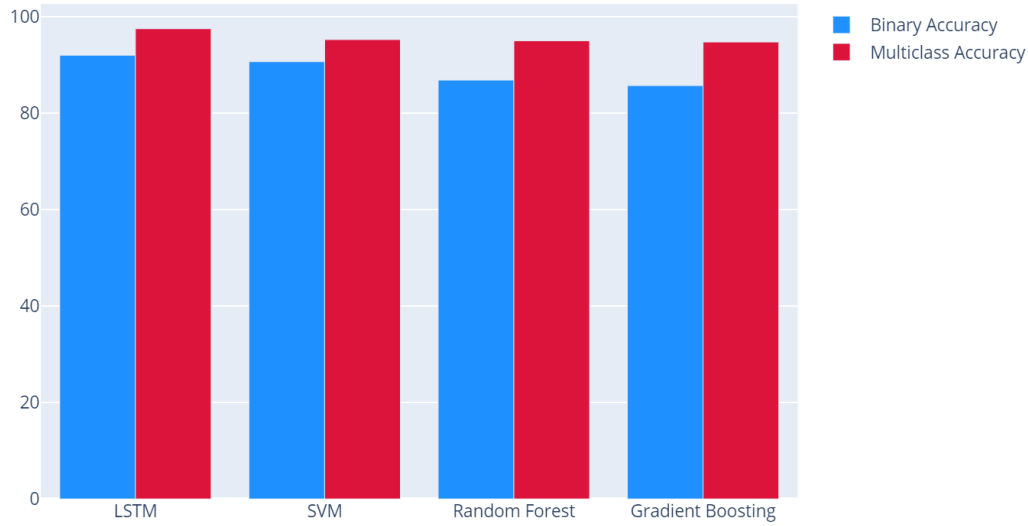


Figure 4.7: Binary's Accuracy vs Multiclass's Accuracy

relatively high prediction probability of 0.46. This explains why, as shown in Figure 4.12, the Gradient Boosting model has the lowest reliability score (0.65) among the models. It is noted that the input text is misclassified as religion-based cyberbullying by machine learning models at varying levels of confidence. This can be

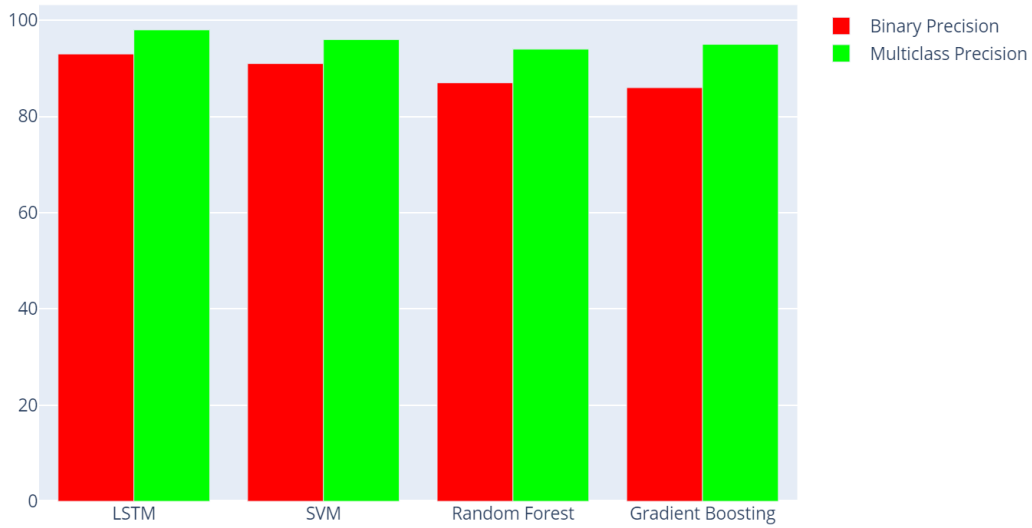


Figure 4.8: Binary's Precision vs Multiclass's Precision

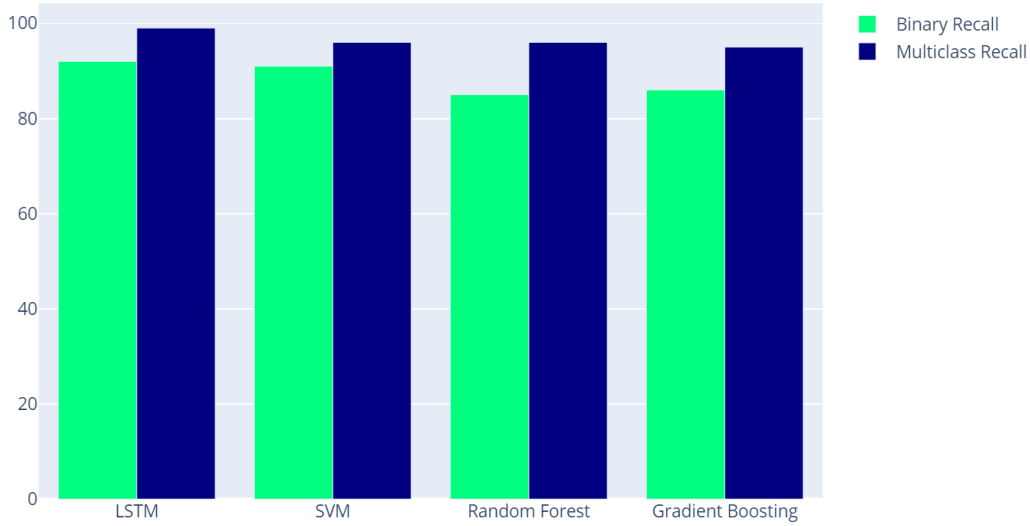


Figure 4.9: Binary's Recall vs Multiclass's Recall

attributed to the large number of records in the training dataset that include certain words ('females'. 'guy', 'sexist') and are labeled as religion-based cyberbullying. In

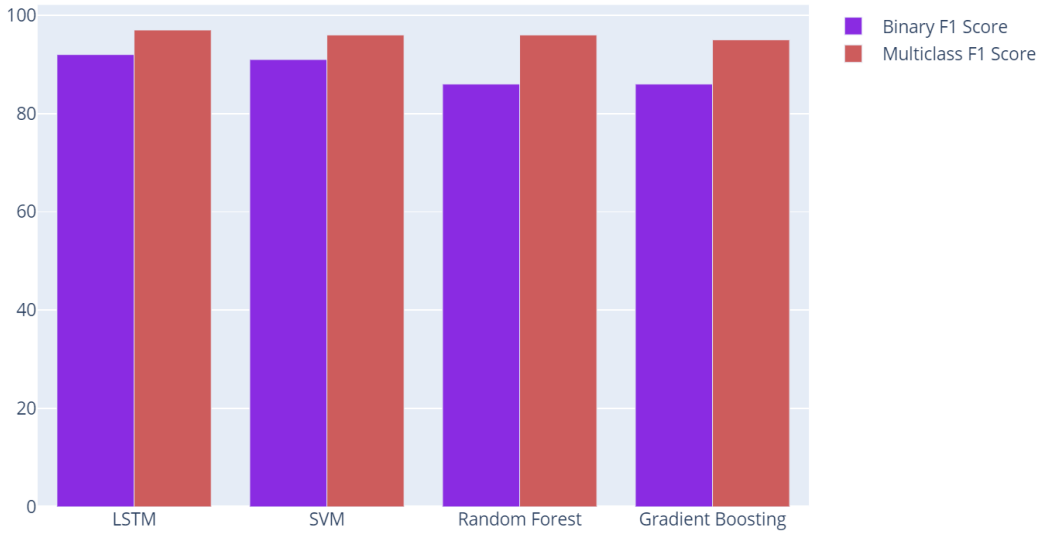


Figure 4.10: Binary's F1-Score vs Multiclass's F1-Score

fact, religions often discuss sexual relationships and women's rights, and individuals may use these issues to attack religions on social media platforms, resulting in religion-based cyberbullying.

4.5.2 Religion-Based Cyberbullying Explanation

In this section, our goal is to compare the explanations for the decisions made by machine learning models when detecting religion-based cyberbullying. To achieve this, we generate an explanation for the following Twitter text: "No, we took it back from Muslims by force. Then what returning you idiot?" After preprocessing, the text is transformed into "take back Muslims force return idiot".

The explanations for RFC, Gradient Boosting, SVM, and LSTM are presented graphically in Figures 4.15, 4.16, 4.17, and 4.18, respectively. As shown in these figures, the machine learning models have arrived at a consensus that the input text

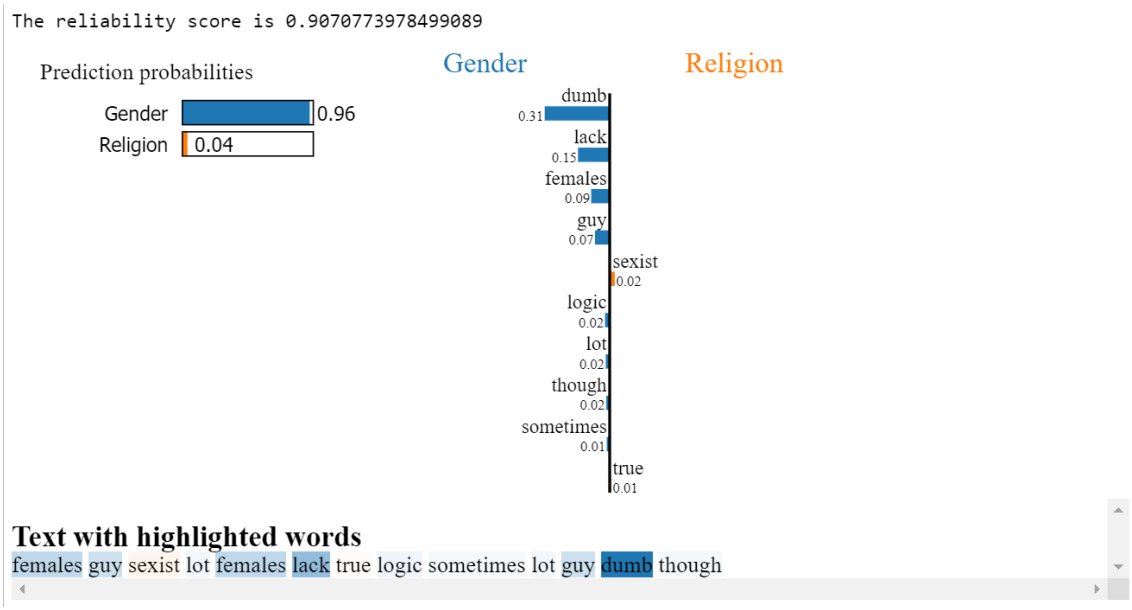


Figure 4.11: Gender-Based Cyberbullying Explanation with Random Forest

indicates religion-based cyberbullying with a high level of confidence, as indicated by their respective prediction probabilities. For example, Figure 4.17 shows that the SVM model exhibits the highest level of confidence, with a prediction probability of 100% that the input text is religion-based cyberbullying. In contrast, Figure 4.16 shows that the Gradient Boosting model exhibits the lowest level of confidence among the models, predicting with a probability of only 84% that the input text is religion-based cyberbullying.

The use of LIME demonstrates that these machine learning models provide similar justifications for their decision-making processes. For instance, all models highlight the words 'idiot' and 'Muslims' as the most influential words affecting the classification of input text as religion-based cyberbullying. Moreover, all machine learning models achieve high-reliability scores, as their justifications are highly consistent with the decision that the input text is religion-based cyberbullying.

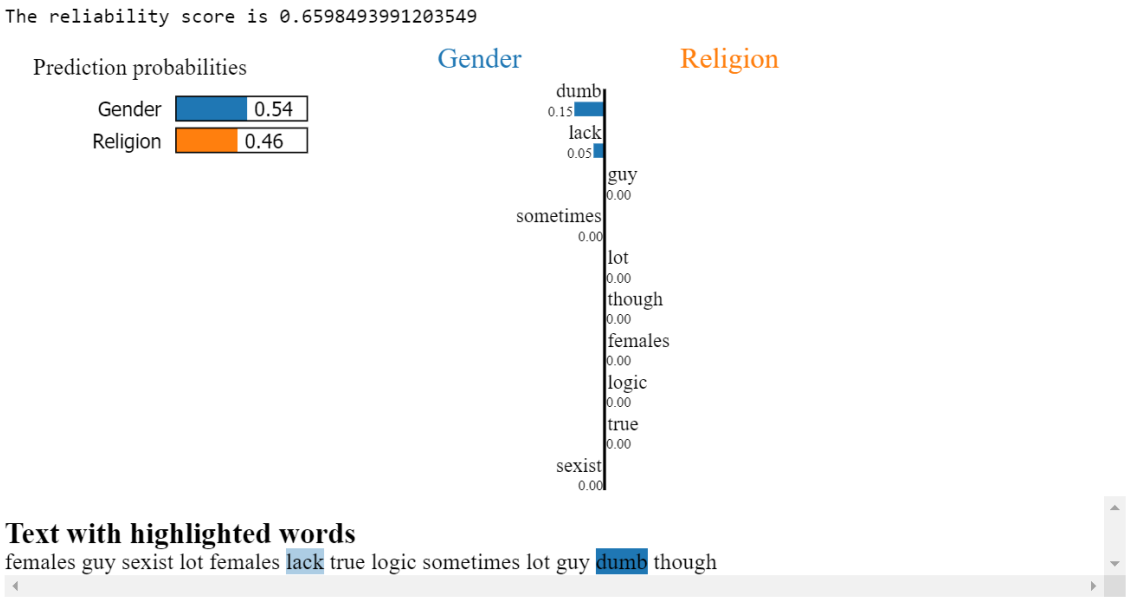


Figure 4.12: Gender-Based Cyberbullying Explanation with Gradient Boosting

It is worth noting that the input text may be misclassified as gender-based cyberbullying by machine learning models, albeit at very low levels of confidence. For instance, as shown in Figure 4.16, the Gradient Boosting model suggests that the input text may be gender-based cyberbullying with a probability of 0.16. This can be attributed to the high number of shared words between religion-based and gender-based cyberbullying in the training dataset, as discussed previously.

4.5.3 Age-Based Cyberbullying Explanation

In this section, our goal is to compare the explanations for the decisions made by machine learning models when detecting age-based cyberbullying. To achieve this, we generate an explanation for the following Twitter text: "Lhdab @ this child here...do ya homework jerk, u new 2d play ground, I'm a school yard bully u new kid". After preprocessing, the text is transformed into "lhdab child ya homework jerk u new play

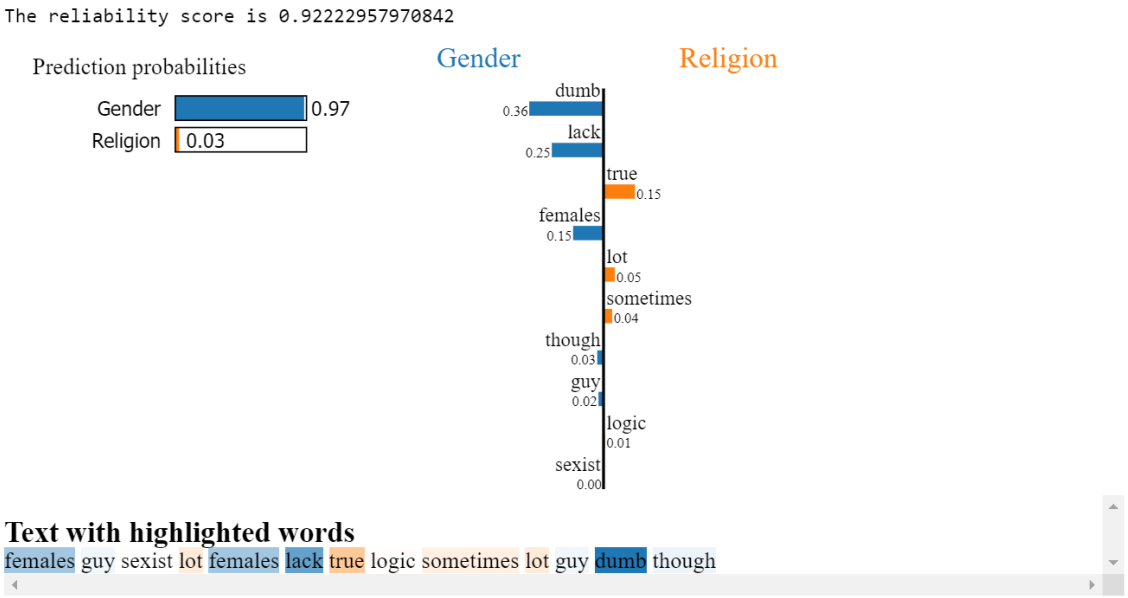


Figure 4.13: Gender-Based Cyberbullying Explanation with SVM

grind school yard bully u new kid".

The graphical explanations for the RFC, Gradient Boosting, SVM, and LSTM models are presented in Figures 4.19, 4.20, 4.21, and 4.22, respectively. These figures demonstrate that the machine learning models have arrived at a consensus that the input text indicates age-based cyberbullying with a high level of confidence, as indicated by their respective prediction probabilities. For example, Figures 4.21 and 4.22 show that the SVM and LSTM models exhibit the highest level of confidence, predicting with a probability of 100% that the input text is age-based cyberbullying. In contrast, Figure 4.20 shows that the Gradient Boosting model exhibits the lowest level of confidence among the models, predicting with a probability of only 73% that the input text is age-based cyberbullying.

LIME shows that machine learning models provide similar justifications for their predictions. For instance, the RFC, SVM, and LSTM models highlight the words

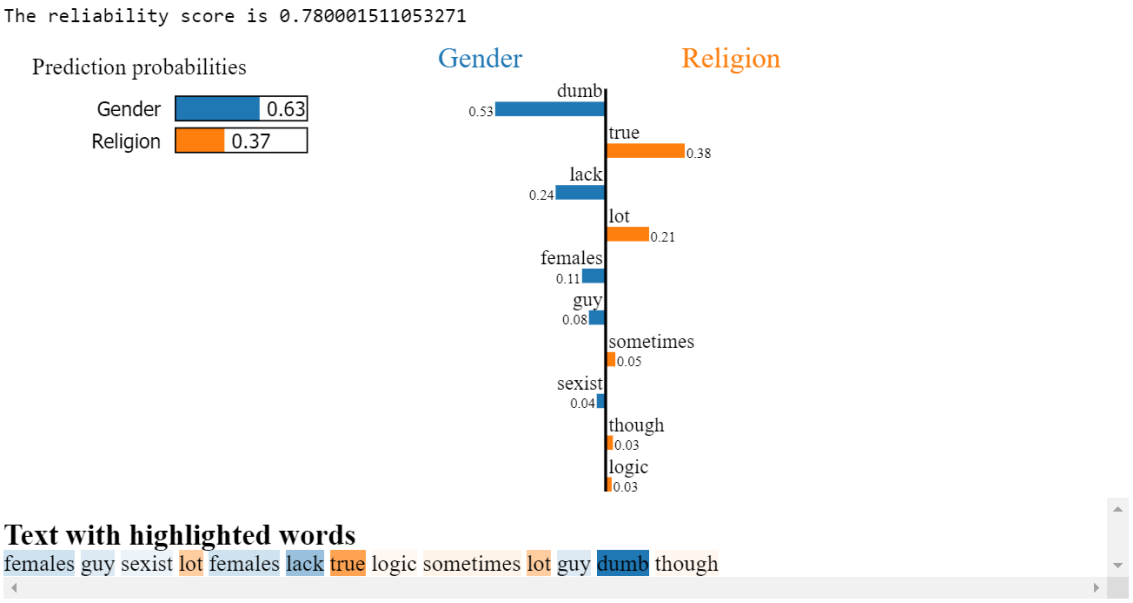


Figure 4.14: Gender-Based Cyberbullying Explanation with LSTM

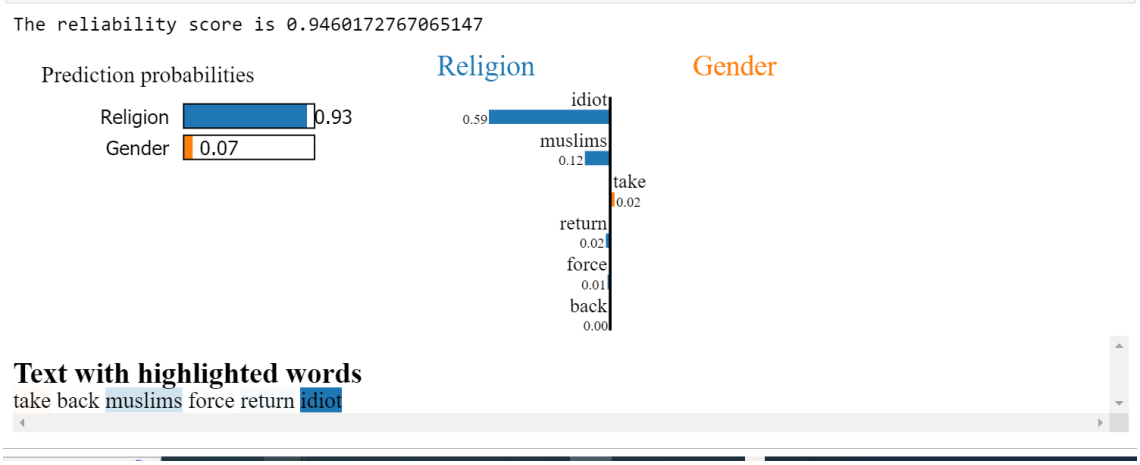


Figure 4.15: Religion-Based Cyberbullying Explanation with Random Forest

'child' and 'homework' as influential words that affect the classification of input text as age-based cyberbullying. In Figure 4.19, the word 'homework' has a weight of 0.08, followed by 'child' at 0.04. These words have similar weights across the RFC, SVM, and LSTM models, as illustrated in Figures 4.19, 4.21, and 4.22.

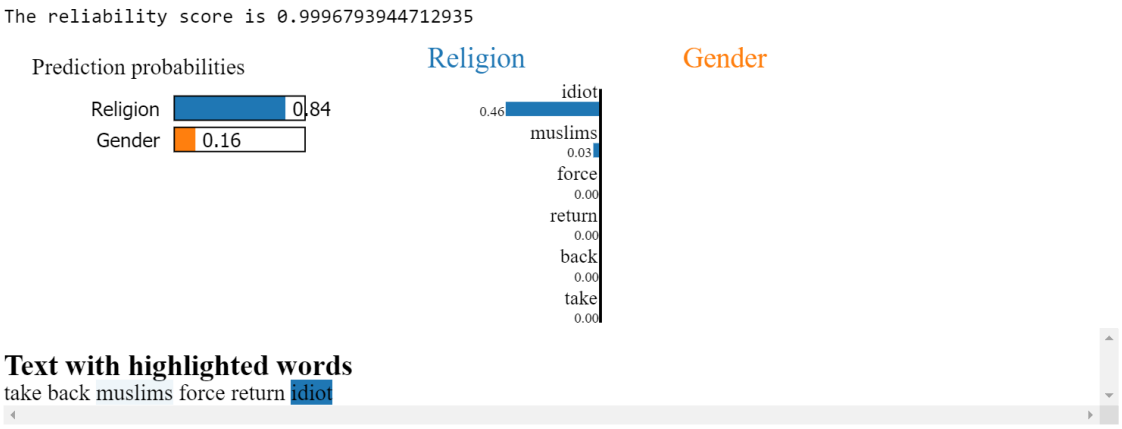


Figure 4.16: Religion-Based Cyberbullying Explanation with Gradient Boosting

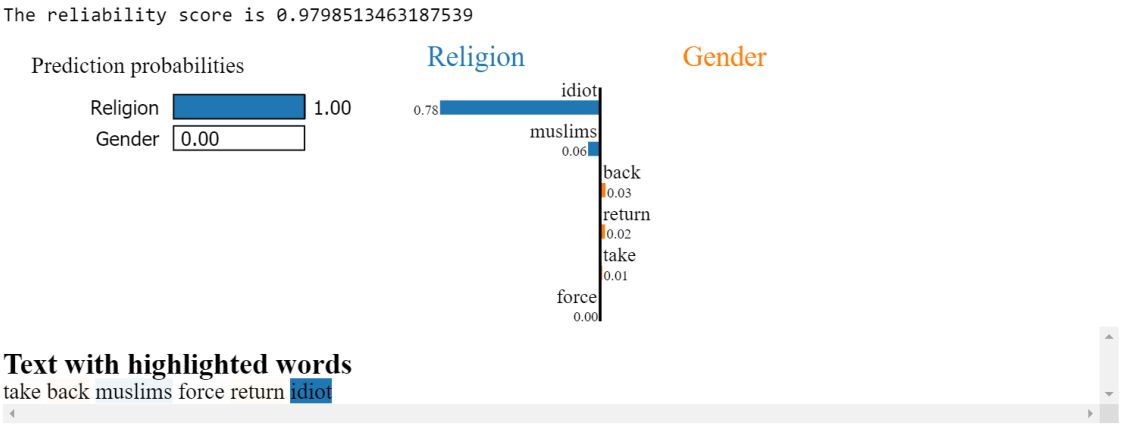


Figure 4.17: Religion-Based Cyberbullying Explanation with SVM

On the other hand, the use of certain words like 'school', 'kid', and 'child' in the text causes the Gradient Boosting model to misclassify the input as ethnicity-based cyberbullying with a relatively high prediction probability of 0.27, as shown in Figure 4.20. This may explain why the Gradient Boosting model has the lowest reliability score (0.75) compared to the other models.

It is important to note that machine learning models may misclassify input text as ethnicity-based cyberbullying with varying levels of confidence. This is likely due to

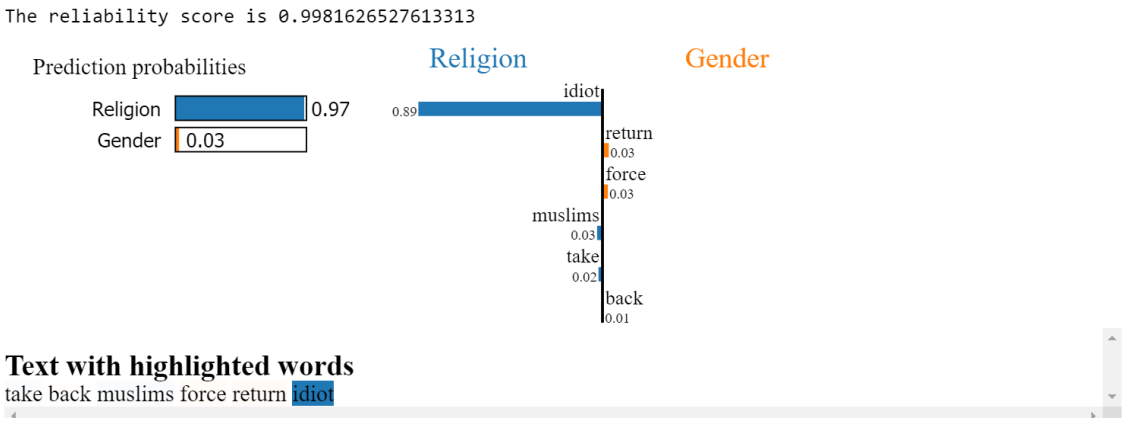


Figure 4.18: Religion-Based Cyberbullying Explanation with LSTM

the large number of records in the training dataset that contain these words and are labeled as ethnicity-based cyberbullying. It is worth noting that most cyberbullying among teenagers in schools is related to ethnicity, so words like 'school' and 'kid' frequently appear in ethnicity-based cyberbullying.

4.5.4 Ethnicity-Based Cyberbullying Explanation

The aim of this section is to compare the explanations provided by the machine learning models when detecting ethnicity-based cyberbullying. To achieve this, we analyze the decision-making process of each model using the following Twitter text: "For example: racism ain't the same as 'speciesism'. Why are you comparing colored people to cows, homie?". After preprocessing the text, we transform it into "example racism speciesism u compare color people cow homie".

The graphical explanations for the RFC, Gradient Boosting, SVM, and LSTM models are presented in Figures 4.23, 4.24, 4.25, and 4.26, respectively. These figures demonstrate that the machine learning models have arrived at a consensus that the input text indicates ethnicity-based cyberbullying with a high level of confidence,

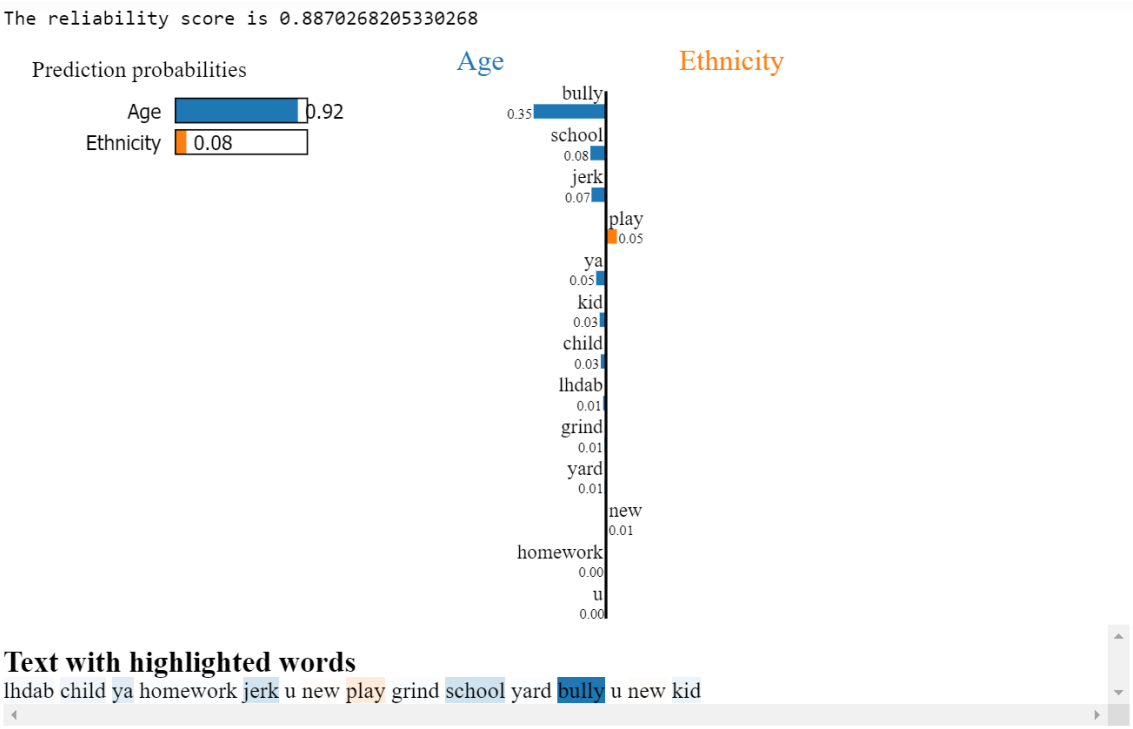


Figure 4.19: Age-Based Cyberbullying Explanation with Random Forest

as indicated by their respective prediction probabilities. For example, Figures 4.26 shows that the LSTM model exhibits the highest level of confidence, predicting with a probability of 100% that the input text is ethnicity-based cyberbullying. In contrast, Figure 4.24 shows that the Gradient Boosting model exhibits the lowest level of confidence among the models, predicting with a probability of only 82% that the input text is ethnicity-based cyberbullying.

According to LIME, machine learning models provide similar justifications to explain the predictions. For instance, all the machine learning models identify the words 'racism', 'people', and 'color' as significant terms that affect the classification of input text as ethnicity-based cyberbullying.

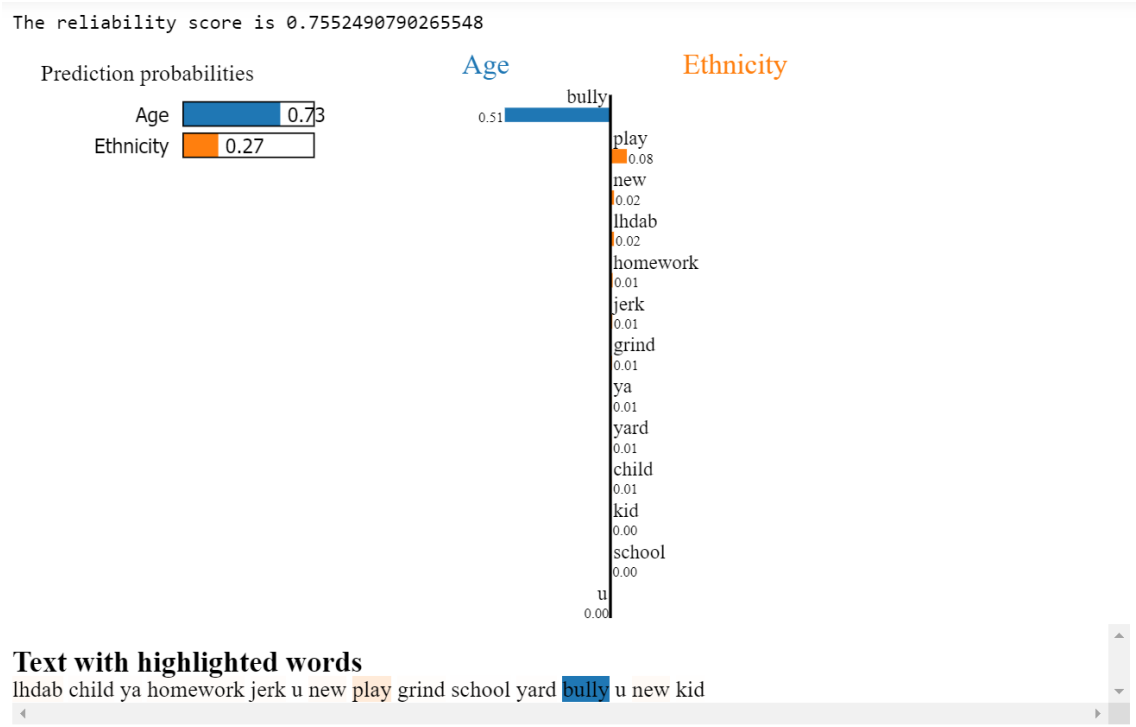


Figure 4.20: Age-Based Cyberbullying Explanation with Gradient Boosting

4.6 Threats to Validity

4.6.1 External Validity

Our cyberbullying detection model is only tested on a publicly available dataset that may not cover all possible variations of language, slang, and communication styles found in real-world cyberbullying texts on social media. As a result, our model may perform well on the dataset but may not generalize well on new, unseen data. Language and communication styles change with time, and a model trained on a static dataset may not be able to adjust to these changes well, limiting its efficiency in more recent cases of cyberbullying. Testing our model on a more diverse dataset that covers different social media and languages would be a good future step to demonstrate the

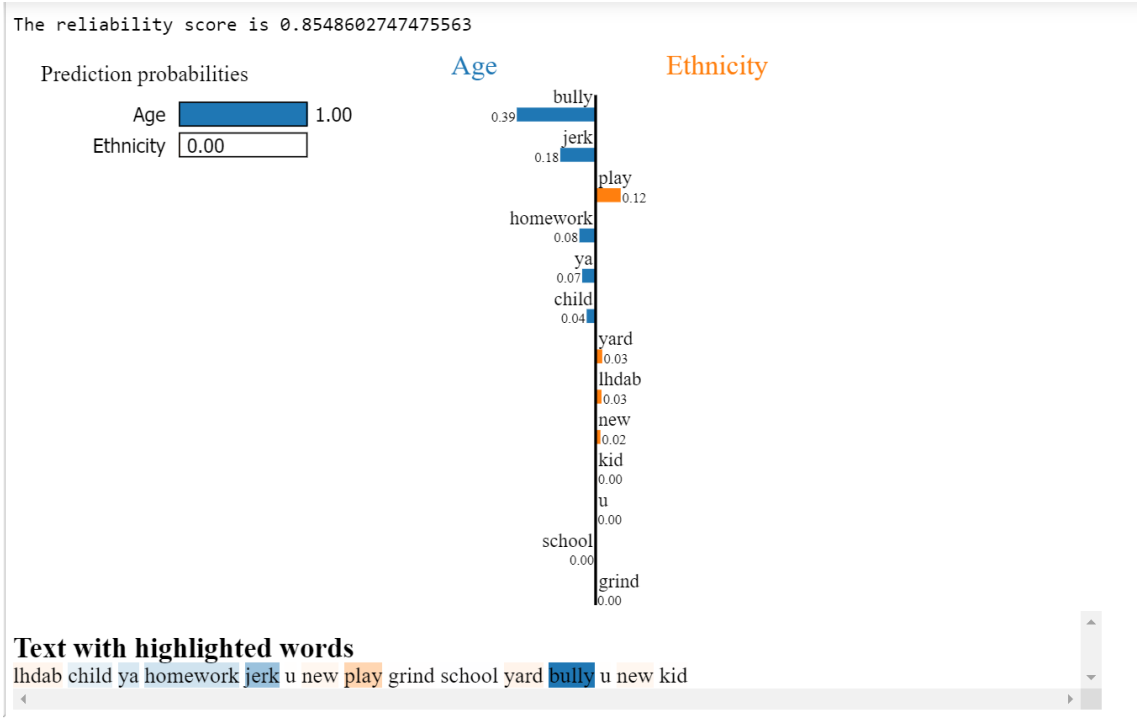


Figure 4.21: Age-Based Cyberbullying Explanation with SVM

model's ability.

The model uses the XAI algorithm, LIME, to explain the binary and multiclass cyberbullying predictions made by the machine learning models. Although LIME is designed to work with different models, the quality and consistency of explanations may vary depending on the model's complexity and decision-making process. It is possible that some models produce more trustworthy explanations than others. This limitation is addressed by analyzing the consistency of the generated explanations with feature importance scores, such as the weights of the words. This can help determine whether the LIME explanations for a certain model are trustworthy. Comparing LIME's explanation with other XAI algorithms, such as SHAP, can help determine the best explanation method for a particular model. The functionality of

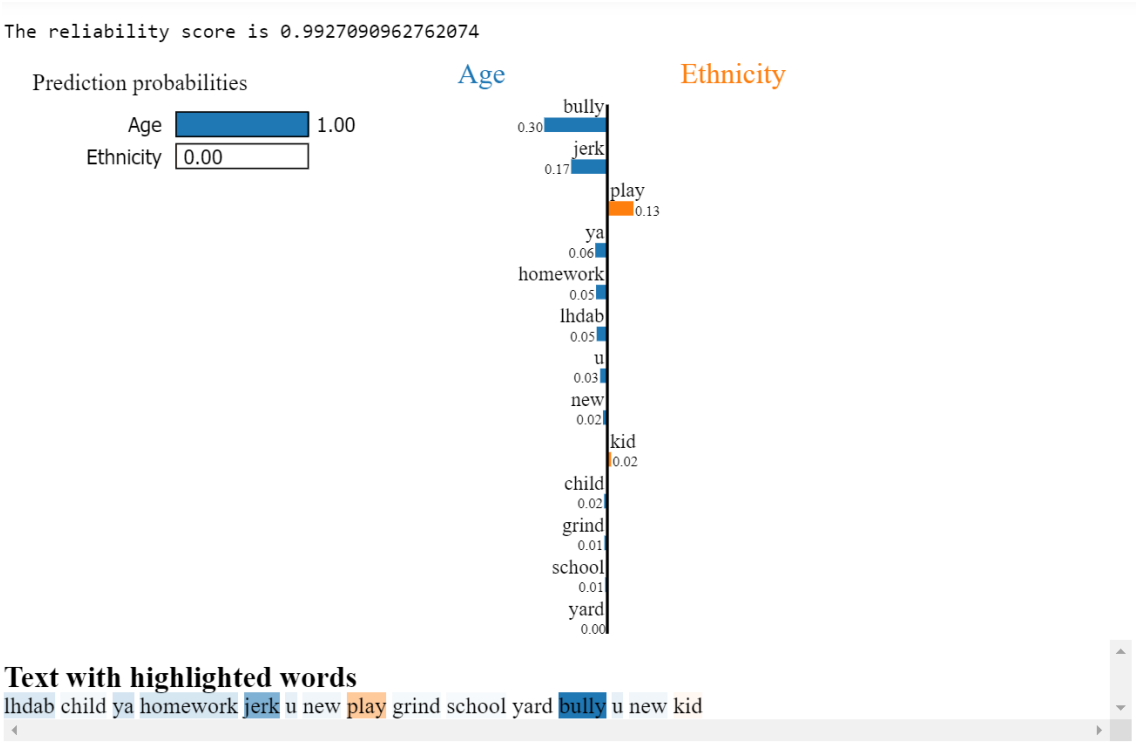


Figure 4.22: Age-Based Cyberbullying Explanation with LSTM

our model can be expanded by combining LIME and SHAP explanations to provide both individual and overall explanations for cyberbullying prediction.

4.6.2 Internal Validity

The chosen NLP tools in our model, sentiment analysis and TF-IDF, may not be sufficient enough to capture all the relevant information for cyberbullying detection, which may limit the performance of our model. We address this issue in our model by combining sentiment analysis and TF-IDF to create an ensemble method for detecting cyberbullying. This ensemble method helps us capture different aspects of a cyberbullying text, such as the frequency of offensive words contributing to the negative sentiment. We also use the ensemble method to train the machine learning

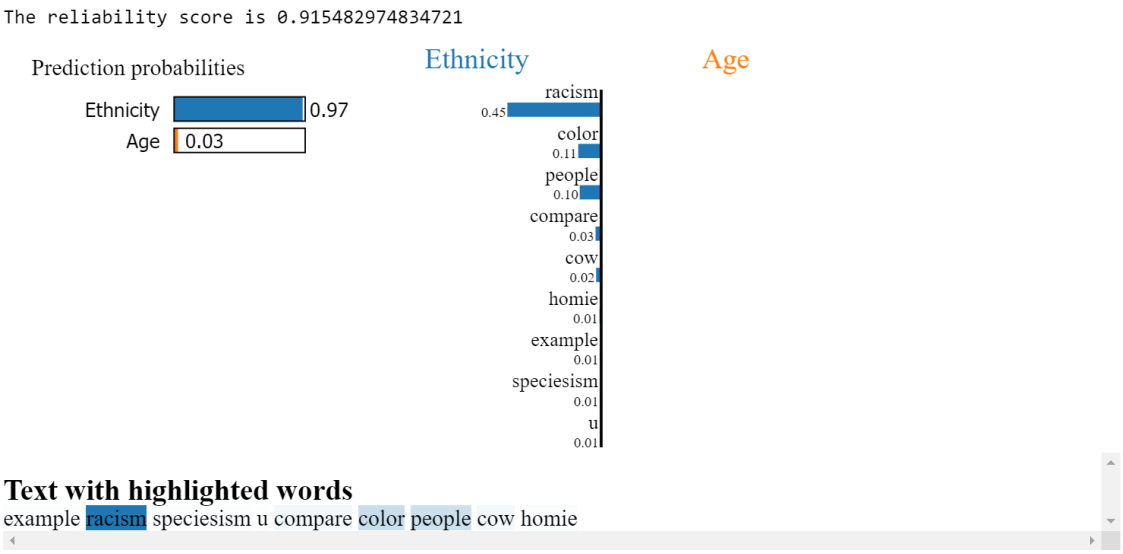


Figure 4.23: Ethnicity-Based Cyberbullying Explanation with Random Forest

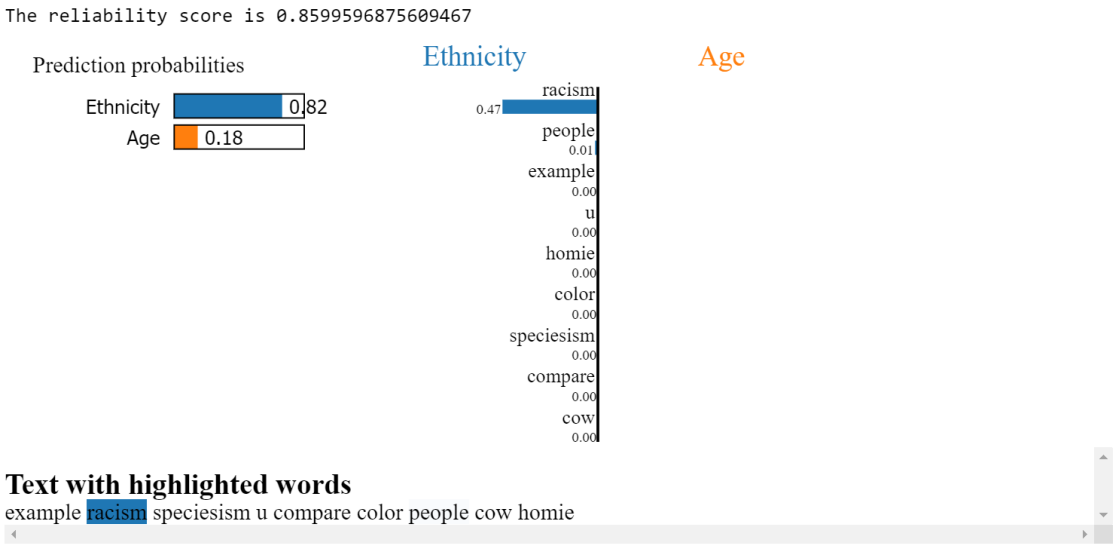


Figure 4.24: Ethnicity-Based Cyberbullying Explanation with Gradient Boosting

models to improve our cyberbullying detection model's performance. In the future, it will be helpful to utilize an additional NLP tool, such as the pre-trained language model BERT [57], to capture the semantic meaning of words in a cyberbullying text

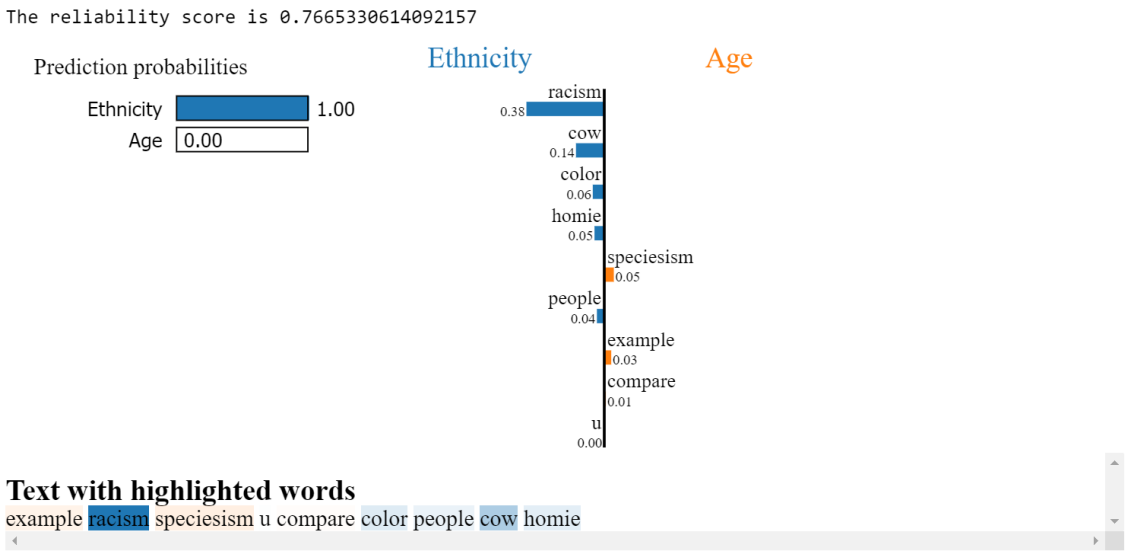


Figure 4.25: Ethnicity-Based Cyberbullying Explanation with SVM

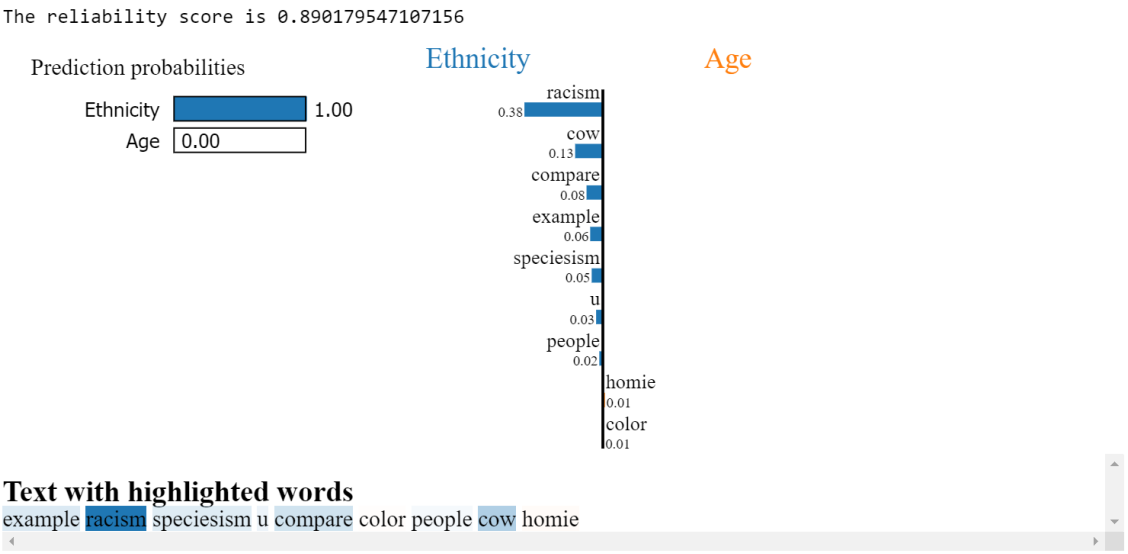


Figure 4.26: Ethnicity-Based Cyberbullying Explanation with LSTM

in our model.

The LIME explanations are based on approximations. Therefore, there may be

differences between the actual reasons behind a machine learning model's cyberbullying prediction and the reasons provided by LIME. This could result in inaccurate explanations. To address this issue, we compare the influential words identified by LIME with those that are considered important for cyberbullying detection to evaluate the quality of LIME explanations. In the future, we can engage domain experts or users in the explanation process by presenting them with the LIME explanations and collecting their feedback on the quality and relevancy of the explanations.

4.7 Summary

This chapter begins by evaluating the performance of machine learning models for binary cyberbullying text predictions using the following evaluation metrics: accuracy, precision, recall, and F-score. It is followed by the LIME explanations for a binary cyberbullying text prediction made by the machine learning models to show the prediction probability, reliability score of a prediction, and weight scores of words.

Next, the performance of machine learning models for multiclass cyberbullying text predictions is provided. We observe that in the case of multiclass cyberbullying text predictions, the accuracy, precision, recall, and F-score have increased for each model compared to binary cyberbullying texts with the help of the XAI algorithm, LIME. The LIME explanations for multiclass cyberbullying predictions are then provided to show the weights of important multiclass cyberbullying words and compare the performances of machine learning models.

Finally, we discuss the external and internal validity of our cyberbullying detection system to show the limitations of our system, how we addressed them in our research, and what steps can be taken in the future to overcome these limitations.

Chapter 5

Conclusion and Future Work

This thesis addresses the challenge of identifying and detecting instances of cyberbullying in social media texts, such as posts, comments, and messages. The problem is approached from three perspectives: 1) How can current research be improved to identify the type of cyberbullying based on characteristics such as religion, gender, age, and ethnicity, given a dataset that provides limited information for classifying texts as cyberbullying or non-cyberbullying? 2) How can current research be improved to enable end-users to easily understand the decisions made by machine learning models, including why victims are attacked by bullies? 3) How can current research be improved to increase the performance of machine learning models, specifically in terms of accuracy?

This thesis proposes and demonstrates the use of an XAI (Explainable Artificial Intelligence) technology called LIME to address those raised issues. Specifically, LIME provides justifications and explanations for why a certain social media text is classified as cyberbullying. These explanations enable end-users to extract and understand information beyond the classification problem. For example, users can

identify the most offensive words used and recognize the reasons and characteristics that incentivize bullies to attack their victims. Additionally, these explanations can be beneficial for social media platforms to develop proactive strategies, such as constructing a filtering AI algorithm that prevents bullies from viewing certain posts.

However, in this thesis, we have gone beyond the traditional use of LIME by solely obtaining explanations. Instead, we have utilized these explanations to provide feedback into the training dataset, following which the machine learning models are retrained on the updated dataset. This idea allows us to automatically detect and identify the type of cyberbullying based on characteristics such as religion, gender, ethnicity, and age, even if the original records in the training dataset do not include such information or features. In our approach, the original records are limited to two classes: cyberbullying and non-cyberbullying. This idea also improves significantly the performance of the machine learning models in terms of classification accuracy.

In this thesis, LIME has been used in conjunction with four machine learning models: RFC, LSTM, SVM, and Gradient Boosting. The simulation results indicate that the LSTM model outperformed the other models in terms of classification accuracy. Furthermore, the RFC model demonstrated the highest level of correspondence between the decisions made and the explanations provided for those decisions. On the other hand, the Gradient Boosting model exhibited the worst performance in terms of classification accuracy and consistency between the justifications and decisions made.

In future work, we will investigate the capabilities of other XAI algorithms such as SHAP, as well as other machine learning models, in detecting cyberbullying in texts. We are also interested in applying the proposed approach to detect and identify cyberbullying in images and videos.

Bibliography

[1] M. A. Al-Garadi, K. D.Varathan, and S. D. Ravana, "Cybercrime detection in online communications: The experimental case of cyberbullying detection in the twitter network," *Computers in Human Behavior*, vol. 63, pp. 433–442, 2016.

[2] D. V. Bruwaene and D. Inkpen, "A multi-platform dataset for detecting cyberbullying in social media," *Language Resources and Evaluation*, vol. 54, pp. 851–874, 2020.

[3] Anastasija Dojchinovska, "18 cyberbullying statistics canada infographics," 2022, https://reviewlution.ca/resources/cyberbullying-statistics-canada/, Last accessed on 2022-08-30.

[4] Monica Anderson, "A majority of teens have experienced some form of cyberbullying," 2018, https://www.pewresearch.org/internet/2018/09/27/a-majority-of-teens-have-experienced-some-form-of-cyberbullying/, Last accessed on 2022-10-24.

[5] Cara Murez, "More cyberbullying, more suicidal thoughts among teens: Study," 2022, https://www.usnews.com/news/health-news/articles/2022-06-28/more-cyberbullying-more-suicidal-thoughts-among-teens-study, Last accessed on 2023-02-04.

[6] M. Khairy, T. M. Mahmoud, and T. Abd-El-Hafeez, "Automatic detection of cyberbullying and abusive language in arabic content on social networks: A survey," *Procedia Computer Science*, vol. 189, pp. 156–166, 2021.

[7] M. Gada, K. Damania, and S. Sankhe, "Cyberbullying detection using LSTM-CNN architecture and its applications," in *2021 International Conference on Computer Communication and Informatics (ICCCI)*, 2021, pp. 1–6.

[8] S. Kadamgode, W. Shi, and J.-P. Corriveau, "Cyberbullying detection using ensemble method," in *3rd International Conference on Data Science and Machine Learning (DSML 2022)*, vol. 12, 2022, pp. 75–94.

[9] S. Singh, K. Kumar, and B. Kumar, "Sentiment Analysis of Twitter Data Using TF-IDF and Machine Learning Techniques," in *2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON)*, vol. 1, 2022, pp. 252–255.

[10] V. Nahar, S. Al-Maskari, X. Li, and C. Pang, "Semi-supervised learning for cyberbullying detection in social networks," in *Databases Theory and Application*, 2014, pp. 160–171.

[11] D. Srivastava and V. Kumar Soni, "A Systematic Review On Sentiment Analysis Approaches," in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 2022, pp. 01–06.

[12] J.-M. Xu, K. S. Jun, X. Zhu, and A. Bellmore, "Learning from bullying traces in social media," in *2012 Conference of the North American Chapter of the*

Association for Computational Linguistics: Human Language Technologies, 2012, pp. 656–666.

[13] H. Dani, J. Li, and H. Liu, "Sentiment informed cyberbullying detection in social media," in *2017 Machine Learning and Knowledge Discovery in Databases*, 2017, pp. 52–67.

[14] K. Dinakar, B. Jones, C.Havasi, H. Lieberman, and R.Picard, "Commonsense reasoning for detection, prevention, and mitigation of cyberbullying," *ACM Transactions on Interactive Intelligent Systems*, vol. 3, pp. 1–30, 2012.

[15] L. Xiang, "Application of an improved TF-IDF method in literary text classification," *Advances in Multimedia*, vol. 2022, p. 10, 2022.

[16] N. M. G. D. Purnamasari, M. A. Fauzi, I. Indriati, and L. S. Dewi, "Cyberbullying identification in twitter using support vector machine and information gain based feature selection," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, pp. 1494–1500, 2020.

[17] R. R. Dalvi, S. Baliram Chavan, and A. Halbe, "Detecting a twitter cyberbullying using machine learning," in *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2020, pp. 297–301.

[18] Z. L. Chia, M. Ptaszynski, F. Masui, G. Leliwa, and M. Wroczynski, "Machine learning and feature engineering-based study into sarcasm and irony classification with application to cyberbullying detection," *Information Processing Management*, vol. 58, p. 102600, 2021.

[19] J. Wu, M. Wen, R. Lu, B. Li, and J. Li, "Toward efficient and effective bullying detection in online social network," *Peer-to-Peer Networking and Applications*, vol. 13, no. 5, pp. 1567–1576, 2020.

[20] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[21] N. Novalita, A. Herdiani, I. Lukmana, and D. Puspandari, "Cyberbullying identification on twitter using random forest classifier," *Journal of Physics: Conference Series*, vol. 1192, no. 1, p. 012029, 2019.

[22] A. Squicciarini, S. Rajtmajer, Y. Liu, and C. Griffin, "Identification and characterization of cyberbullying dynamics in an online social network," in *2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2015, pp. 280–285.

[23] D. Chatzakou, N. Kourtellis, J. Blackburn, E. De Cristofaro, G. Stringhini, and A. Vakali, "Mean birds: Detecting aggression and bullying on twitter," in *Proceedings of the 2017 ACM on Web Science Conference*, 2017, p. 13–22.

[24] V. Balakrishnan, S. Khan, and H. R. Arabnia, "Improving cyberbullying detection using twitter users' psychological features and machine learning," *Computers Security*, vol. 90, p. 101710, 2020.

[25] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers in Neurorobotics*, vol. 7, p. 21, 2013.

[26] A. Muneer and S. M. Fati, "A comparative analysis of machine learning techniques for cyberbullying detection on twitter," *Future Internet*, vol. 12, no. 11, p. 187, 2020.

[27] M. R. Kurniawanda and F. A. Tobing, "Analysis sentiment cyberbullying in instagram comments with xgboost method," *International Journal of New Media Technology*, vol. 9, no. 1, p. 28–34, 2022.

[28] T. Joachims, "Text Categorization with Support Vector Machines: Learning with many relevant features," in *Machine Learning: ECML-98*, 1998, pp. 137–142.

[29] H. K. Sharma, K. Kshitiz, and Shailendra, "NLP and machine learning techniques for detecting insulting comments on social networking platforms," in *2018 International Conference on Advances in Computing and Communication Engineering (ICACCE)*, 2018, pp. 265–272.

[30] J. Bhagya and P. S. Deepthi, "Cyberbullying detection on social media using SVM," in *Inventive Systems and Control*, 2021, pp. 17–27.

[31] W. K. Sari, D. P. Rini, and R. F. Malik, "Text classification using Long Short-Term Memory," in *2019 International Conference on Electrical Engineering and Computer Science (ICECOS)*, 2019, pp. 150–155.

[32] S. Wang, W. Zhou, and C. Jiang, "A survey of word embeddings based on deep learning," *Computing*, vol. 102, no. 3, pp. 717–740, 2019.

[33] D. Dessì, D. R. Recupero, and H. Sack, "An assessment of deep learning models and word embeddings for toxicity detection within online textual comments," *Electronics*, vol. 10, no. 7, 2021.

[34] A. Dass and D. K. Daniel, "Cyberbullying detection on social networks using LSTM model," in *2022 International Conference on Innovations in Science and Technology for Sustainable Development (ICISTSD)*, 2022, pp. 293–296.

[35] S. Agrawal and A. Awekar, "Deep learning for detecting cyberbullying across multiple social media platforms," in *Advances in Information Retrieval*, 2018, pp. 141–153.

[36] A. Al-Hassan and H. Al-Dossari, "Detection of hate speech in arabic tweets using deep learning," *Multimedia Systems*, vol. 28, pp. 1963–1974, 2021.

[37] S. Balakrishna, Y. Gopi, and V. K. Solanki, "Comparative analysis on deep neural network models for detection of cyberbullying on social media," *Ingeniería Solidaria*, vol. 18, no. 1, p. 1–33, 2022.

[38] A. Maranhão, "Cyberbullying dataset," 2022, https://www.kaggle.com/datasets/andrewmvd/cyberbullying-classification, Last accessed on 2023-03-14.

[39] J. Camacho-Collados and M. T. Pilehvar, "On the role of text preprocessing in neural network architectures: An evaluation study on text categorization and sentiment analysis," in *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 2018, pp. 40–46.

[40] S. Bird, E. Klein, and E. Loper, *Natural language processing with python: Analyzing text with the natural language toolkit*. O'Reilly, 2009.

[41] G. Nicolai and G. Kondrak, "Leveraging inflection tables for stemming and lemmatization." *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1138–1147, 2016.

[42] A. Muhammad and A. Dahiru, "Lexicon-based sentiment analysis of web discussion posts using SentiWordNet," *Journal of Computer Science and Its Application*, vol. 26, no. 2, p. 1, 2020.

[43] V. Nahar, S. Unankard, X. Li, and C. Pang, "Sentiment Analysis for effective detection of cyber bullying," in *APWeb 2012: Web Technologies and Applications*, vol. 7235, 2012, pp. 767–774.

[44] A. Dewani, M. A. Memon, and S. Bhatti, "Cyberbullying detection: Advanced preprocessing techniques amp; deep learning architecture for roman urdu data," *Journal of Big Data*, vol. 8, no. 160, 2021.

[45] N. Cristianini and E. Ricci, *Support Vector Machines*. Springer US, 2008.

[46] k.Vijayaprabakaran and K.Sathiyamurthy, "Towards activation function search for long short-term model network: A differential evolution based approach," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 6, Part A, pp. 2637–2650, 2022.

[47] R. C. Staudemeyer and E. R. Morris, "Understanding LSTM - a tutorial into Long Short-Term Memory Recurrent Neural Networks," *ArXiv*, vol. abs/1909.09586, 2019.

[48] J. Manokaran and G. Vairavel, "Giwrf-smote: Gini impurity-based weighted random forest with smote for effective malware attack and anomaly detection in iot-edge," *Smart Science*, pp. 1–17, 2022.

[49] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?" explaining the predictions of any classifier," in *KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 8 2016, pp. 1135–1144.

[50] D. Mardaoui and D. Garreau, "An analysis of LIME for text data," in *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Banerjee and K. Fukumizu, Eds., vol. 130. PMLR, 13–15 Apr 2021, pp. 3493–3501. [Online]. Available: https://proceedings.mlr.press/v130/mardaoui21a.html

[51] Wikitionary-ethnic, "English ethnic slurs," https://en.wiktionary.org/w/index.php?title=Category\%3AEnglish_ethnic_slurs&pagefrom=ROUNDEYE\%0Aroundeye#mw-pages, Last accessed on 2023-03-11.

[52] Wikitionary-religious, "English religious slurs," https://en.wiktionary.org/wiki/Category:English_religious_slurs, Last accessed on 2023-03-11.

[53] Wikipedia, "Sex- and gender-related slurs," Nov 2022, https://en.wikipedia.org/wiki/Category:Sex-_and_gender-related_slurs, Last accessed on 2023-03-11.

[54] Urban-Dictionary, "Cuss words," https://www.urbandictionary.com/define.php?term=cuss+words, Last accessed on 2023-03-11.

[55] M. bierner, "Urban-dictionary-word-list," https://github.com/mattbierner/urban-dictionary-word-list, Last accessed on 2023-03-11.

[56] B. A. H. Murshed, J. Abawajy, S. Mallappa, M. A. N. Saif, and H. D. E. Al-Ariki, "DEA-RNN: A hybrid deep learning approach for cyberbullying detection in twitter social media platform," *IEEE Access*, vol. 10, pp. 25 857–25 871, 2022.

[57] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the*

2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, 2019, pp. 4171–4186.