Bystander Detection: Automatic Labeling Techniques using Feature Selection and Machine Learning

Anamika Gupta[1], Khushboo Thakkar[2], Veenu Bhasin[3], Aman Tiwari[4], Vibhor Mathur[5]
S.S. College of Business Studies, University of Delhi, India[1,2,4,5]
P.G.D.A.V. College, University of Delhi, India[3]

***Abstract*—A hostile or aggressive behavior on an online platform by an individual or a group of people is termed as cyberbullying. A bystander is the one who sees or knows about such incidences of cyberbullying. A defender who intervenes can mitigate the impact of bullying, an instigator who accomplices the bully, can add to the victim's suffering, and an impartial onlooker who remains neutral and observes the scenario without getting engaged. Studying the behavior of Bystanders role can help in shaping the scale and progression of bullying incidents. However, the lack of data hinders the research in this area. Recently, a dataset, CYBY23, of Twitter threads having main tweets and the replies of Bystanders was published on Kaggle in Oct 2023. The dataset has extracted features related to toxicity and sensitivity of the main tweets and reply tweets. The authors have got manual annotators to assign the labels of Bystanders' roles. Manually labeling bystanders' roles is a labor-intensive task which eventually raises the need to have an automatic labeling technique for identifying the Bystander role. In this work, we aim to suggest a machine-learning model with high efficiency for the automatic labeling of Bystanders. Initially, the dataset was re-sampled using SMOTE to make it a balanced dataset. Next, we experimented with 12 models using various feature engineering techniques. Best features were selected for further experimentation by removing highly correlated and less relevant features. The models were evaluated on the metrics of accuracy, precision, recall, and F1 score. We found that the Random Forest Classifier (RFC) model with a certain set of features is the highest scorer among all 12 models. The RFC model was further tested against various splits of training and test sets. The highest results were achieved using a training set of 85% and a test set of 15%, having 78.83% accuracy, 81.79% precision, 74.83% recall, and 79.45% F1 score. Automatic labeling proposed in this work, will help in scaling the dataset which will be useful for further studies related to cyberbullying.**

***Keywords*—Bystanders; cyberbullying; machine learning; defender; instigator; impartial; toxicity; twitter**

I. Introduction

With the emergence of technology in this digital era the dynamics of human connection have changed. Social media platforms have evolved into incredible tools for connecting individuals from all over the world. However, some individuals use it positively while others engage in terrible conduct on social media. The destructive phenomenon of cyberbullying has emerged as a result of the rise of social media platforms [1]. As our lives grow more entwined with the virtual domain, the frequency and consequences of cyberbullying have caught the interest of scholars, educators, and lawmakers.

Bullying is defined as a recurring pattern of hostile or aggressive behavior carried out by an individual or group that meets three criteria: repetition, intent to harm, and lack of authority [2]. The major actors engaged in bullying irrespective of the circumstances in which it occurs are the perpetrator (bully), the victim, and bystanders. Bystanders in the cyberbullying landscape might be considered passive witnesses, which may involve strangers, who are often lured into the online chaos. They have the potential to either perpetuate or mitigate the trauma of victims. Bystanders have the potential to make a positive impact in bullying situations. Victims feel less worried and disappointed when they are surrounded by compassionate peers. Bystanders are present during bullying occurrences 80% of the time, and when they react, the bullying stops in 57% of cases within 10 seconds.

Statistics highlight a harsh reality, emphasizing the importance of acknowledging and addressing cyberbullying. According to recent surveys, an enormous percentage of people of different ages have been victims of internet abuse. Moreover, the findings provide a comprehensive picture, emphasizing the frequency of cyberbullying. Many studies use Twitter as one of the most popular data sources to identify cyberbullying as it is the most popular social networking site where cyberbullying is prevalent because of its constant conversation atmosphere which allows users to openly express their emotions, thoughts, and opinions [3].

Children and teenagers are more familiar with the internet nowadays than ever before, at younger ages. This pattern has given rise to a major concern of cyberbullying [4]. Cyberbullying has a significant impact on victims both physically and psychologically. Bullying can cause depression, anxiety, loneliness, dejection, low self-esteem, anger, self-harming behavior, alcohol and drug usage, and engagement in violence or crime. Physical health suffers as well, resulting in headaches, sleeplessness, abdominal pain, food disorders, and nausea. Cyberbullying has also shown long-term effects on victims, causing stress, continuous misery, sleep difficulties, and even issues like hunger [5].

II. Background and Related Work

To identify bullying, an annotation technique [6] was created to recognize textual aspects of cyberbullying, which includes posts by bullies and responses from victims and the audience. The fundamental goal of [6] research is to acquire an understanding of the language aspects of cyberbullying. This is accomplished in two stages by gathering and annotating a dataset. A harmfulness score is calculated for each post in the first phase to determine whether it is part of a cyberbullying incident. If that's the case, annotators divide the authors' roles into four categories: harasser, victim, bystander defender, and bystander assistant. A binary classifier for each fine-grained

bullying category has been built by the end. Additional features like semantic information were not explored in this research.

The study discovered that the spread of hatred from the primary posts to the replies significantly impacts how annotators identify a thread, frequently leading to reclassification as bullying rather than plain aggression [7][8]. An examination of the entire thread assists annotators in understanding the intent behind the use of specific phrases, which may have different interpretations depending on the context [9]. This finding is consistent with earlier research emphasizing the impact of bystander behavior in online environments. Bystanders' reactions are socially influenced and can be formed by their interactions with offensive comments, resulting in peer pressure and antisocial conduct. The study emphasizes the complex dynamics of online interactions, namely the involvement of bystanders in contributing to the overall classification of content as bullying. The study discovered that the spread of hatred from the primary posts to the replies significantly impacts how annotators identify a thread, frequently leading to reclassification as bullying rather than plain aggression. [7][8] An examination of the entire thread assists annotators in understanding the intent behind the use of specific phrases, which may have different interpretations depending on the context [9]. This finding is consistent with earlier research emphasizing the impact of bystander behavior in online environments. Bystanders' reactions are socially influenced and can be formed by their interactions with offensive comments, resulting in peer pressure and antisocial conduct. The study emphasizes the complex dynamics of online interactions, namely the involvement of bystanders in contributing to the overall classification of content as bullying [7][8].

The work done by [10] focuses on two objectives one is to detect cyberbullying as a binary classification problem and to detect participant roles as a multi-class classification problem. In simple terms, the focus is on evaluating the performance of models that could classify whether the post is cyberbullying-related and if it is the prediction of author's role is done. But there is a need for a more comprehensive and integrated approach that goes beyond individual posts to capture the dynamics of entire discussions in the context of cyberbullying.

While [11] contains two cyberbullying corpora in Dutch and English language. Both are manually annotated with bullying types and participant roles: harasser/bully - the individual who initiates the harassment, Victim - the one who is harassed, Bystander-Assistant: someone who assists the harasser. Bystander-defender:a person who supports the victim. This dataset has a serious problem of imbalance in the data. As "Bystander-Assistant" was the minority class, so the "Bystander-Assistant" was merged with the "Harasser" class to reduce the skew. However, there was still a large amount of imbalance between the "Harasser", "Victim" and "Defender" classes, and between "Bullying" and "No Bullying" in both English and Dutch Corpus which could negatively affect the machine learning corpus. Table II summarizes the related work in this area.

As concluded, there are many datasets available in the field of cyberbullying research on Twitter. Previous studies on cyberbullying detection as mentioned in Table I on Twitter relied on datasets labeled based on individual tweets, failing to capture the complexities of cyberbullying incidents. Labeling the roles of bystanders is a time-consuming job, especially when examining Twitter threads with a significant number of replies, as it demands a thread-by-thread approach thereby creating a need to automate the labeling techniques.

The uniqueness of the dataset [12], [13], [14] used in this research is the inclusion of labels for bystanders' roles and aggressiveness level of Cyberbullying. Many of the existing datasets solely focus on labeling the main post lacking information about the participants involved such as Bystanders. To the best of our knowledge, this dataset is different from the existing datasets. It contains 112 Twitter threads including the main post and the replies on that post totalling around 639 tweets. It also includes the primary tweets and bystander replies. These threads are grouped by conversation ID. By incorporating efficient machine learning models on this dataset better classification can be done leading to a deeper understanding of real-world scenarios [13], [14].

Through the Literature Survey, it can be said that there are not many Twitter datasets available where bystander roles in Cyberbullying are classified. The dataset used here [12], [13], [14] contains multiple types of Bystander roles such as defender, instigator, impartial, or other. It also consists of multi-class labels either as bullying with high aggression, bullying with low aggression, or aggression without indication of bullying.

The rest of the paper is organized as follows: Section II-A presents the motivation and objectives of the proposed work. Section III explains the methodology of the research. Experiments with results and their analysis are discussed in the Section IV followed by conclusions and suggestions for future work in Section V.

TABLE I. PUBLICLY AVAILABLE DATASETS FOR CYBERBULLYING

Data Source	Data size	Data Language	Data Gathering Tools
ASKfm[6]	91,370 Dutch posts	Dutch	GNU Wget software
ASKfm[10]	-	English	AMICA
Facebook[15]	100 comments	English	
ASKfm[4] [11]	113,698 English, 78,387 Dutch	English and Dutch	GNU Wget software
Twitter[16]	79,799 conversations with 528,041 tweets	English	Twarc

A. Motivation

The risk of cyberbullying is increasing year by year due to increased access to technology, low-cost internet connections, and the leaders enthusiastically pursuing and pushing the dream of "Digital India," making its assessment and prevention even more crucial. The vast majority of people now have access to the Internet. The children and teenagers are the most susceptible members, as they are driven into cyberspace before they are psychologically capable of making sense of it. According to Microsoft's Global Youth Online Behaviour Survey, India ranks third in cyberbullying, with 53% of respondents, primarily youngsters, admitting to have experienced online bullying, trailing only China and Singapore.

TABLE II. CYBERBULLYING DETECTION, AND BULLYING TYPES

Characteristics	Preprocessing steps	Classifier	Technique	Classification
Bag of words, polarity based on sentiment lexicon features [6]	Tokenization, lemmatization and PoS-tagging	Binary Classifier	SVM	Harasser, victim and bystander
An ensemble model is extended with a pre-trained BERT embedding layer, hidden neural layer, and a softmax output layer [10]	Replacing slang words, abbreviations, decoding emoticons, punctuations removal, upper to lower case, tokenization and special token additions	Binary Classifier	Ensemble model	Harasser, Victim, Bystander defender, Bystander assistant
Latent Semantic Analysis, multitask multimodality Gated Recurrent Unit, and Dirichlet Multinomial Mixture are applied to detect cyberbullying [15]	Tokenization, lemmatization, stemming, removing special characters and stop words,	Random Forest	Latent semantic analysis and feature extraction	Denigration, Trickery, Flaming, and Cyberstalking
Discovering bystander effect from the negative correlation between the number of Twitter users in the conversation before a toxic tweet was sent and the number of users who responded to the toxic tweet in a non-toxic manner. [16]	tweets with only links, images, and videos were discarded	-	Multivariate regression analysis, Poisson regression model, linear regression model	Bystanders
Multiclass classification to determine cyberbullying with Participant role detection. Investigating feature-engineered single and ensemble classifier setups and transformer-based pre-trained language models (PLMs) [11]	Tokenization, lemmatization and part-of-speech-tagging	Linear classification, Voting classifier, Cascading classifier	SVM, Logistic regression, passive-aggressive, SGD Random BL, Majority BL	Harasser, Victim, Bystander defender, Bystander assistant

Bystanders play an important role in dealing with cyberbullying situations where they can change the dynamics of relationships. They can respond in three ways: by replicating the perpetrator's toxic behavior, by interfering with the toxic talk and sticking up for the victim, or by just observing the unfolding events. However, the mechanisms of bystander behavior in cyberspace in response to hate speech are complex. This complication emerges because the existence of other internet users may reduce one's sense of obligation to interfere, expecting that someone else will do so. Bystanders in smaller groups, on the other hand, feel a larger need to intervene in cases of cyberbullying [17].

Most of the datasets that are available publicly do not emphasize any information related to the Bystander roles in Cyberbullying. Considering the effect of the bystanders, it is important to classify its role. The motive is to explore and potentially implement automatic labeling techniques for the dataset CYBY23 [12]. The integration of automated labeling techniques in the dataset CYBY23 [12] helps to enhance the dataset's scalability and usability for future studies in cyberbullying research. The overarching goal is to contribute to the advancement of research in the field, offering insights that can foster a healthier online environment.

B. Objective

In this work, we aim to suggest a highly efficient technique for

1) Automated labeling of bystander roles in cyberbullying tweets.
2) Finding out the most effective features extracted from the text of the tweets.

For the above objectives, we will deploy several machine learning models and experiment with various pre-processing, and feature selection techniques to discover the most efficient one among those.

III. METHODOLOGY AND PROPOSED MODEL

In this section, the methodology of our research work is described. Flow chart for the same is given in Fig. 1. The Major steps are listed below:

1) Data Ingestion: The dataset, CYBY23, was downloaded from the Kaggle website [13], [12].
2) Data Pre-processing: Initially, the imbalance of the data was removed by using the SMOTE technique [18]. Further, data was pre-processed to make it suitable for machine learning models. The features of the main tweet were augmented with those of reply tweets and some unwanted features were removed. Categorical features were converted to numeric values.
3) Deployment of Machine Learning Models: Twelve machine-learning models [19] were deployed on the pre-processed data of Bystanders. The parameters of all the models were hypertuned to give their best performance. Pycaret library of Python [1] was used for this purpose. The models were evaluated based on accuracy, precision, recall, and F1 score metrics.
4) Experiments with Feature Selection: Next, various combinations of feature sets were experimented with like Toxicity features only (extracted from Perspective API [2]), Sensitivity features only (extracted from TextBlob [3]), and combinations of these features. Further, highly correlated features and less relevant features were removed to judge the performance of machine learning models.

Finally, a machine learning model having best accuracy and F1 score was recommended for automatic labeling of Bystanders role. The automation of Bystanders role detection will help in the early detection of cyberbullying cases and reduce their number to a greater extent.

Each of the steps involved in the process is explained below in detail:

[1] https://pycaret.org
[2] https://perspectiveapi.com/
[3] https://textblob.readthedocs.io/en/dev/

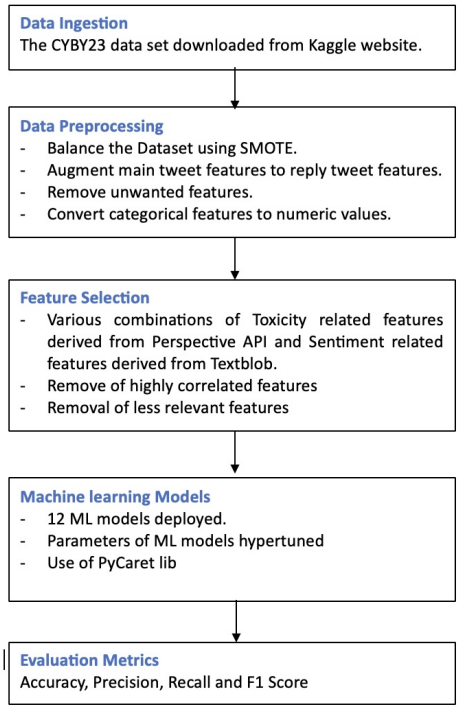

Fig. 1. Flow chart of methodology

A. Dataset Description

The dataset related to bystanders was downloaded from Kaggle [12]. Alfurayj et al. [13] used Twitter API to extract 1024 tweets from January 2022 to January 2023. 150 tweet threads were collected. Information such as the date of the tweet, tweet ID, screen name of the user and user ID associated with the tweet, number of likes & retweets, and text of the tweet was downloaded. Religion, ethnicity, sarcasm, and racial orientation were among the keywords and hashtags used to crawl this information, which could lead to harassment remarks. A manual annotation process for the labeling of Bystanders was used. Annotators followed the guidelines given in [20] and assessed the aggressiveness of individual tweets, identified bystander roles in replies, and made higher-level judgments about the overall aggressiveness of the thread after considering the main post, replies, and bystander roles. Following the annotation process, threads lacking agreement from at least five annotators were eliminated, reducing the tweets to 639. The dataset, meeting the criteria for a good dataset, contained a minimum of 10% to 20% bullying cases, with cyberbullying with high aggression representing only 11.6%. Instigators were notably high in both bullying categories. The investigation focused on bystander contagion risk, with a higher prevalence of instigators associated with instances of bullying, as evidenced by the dataset. They realized the need for the automation of annotation for labeling of Bystanders' role because of the labor-intensive nature of manual annotation and hence a dataset, named CYBY23, was uploaded on the Kaggle website [12] for public use. CYBY23 dataset had the Twitter threads containing both the main posts and the replies from Bystanders. Each tweet had the text of the tweet along with certain general features of the tweets. Further, they extracted the Toxicity features using Perspective API and sentiment features using TextBlob for each tweet. There were 639 tweets in the dataset with the labels of bystanders' roles (manually annotated).

So, the dataset, CYBY23 [12], had six general features, namely, tweet_id, reply_id, text , created_at, favorite_count, retweet_count for each tweet. Six features were derived from Perspective API , namely, Insult, Threat, Identity_Attack , Profanity ,Toxicity , and Severe_Toxicity, and three features were derived from TextBlob , namely, polarity, subjectivity, and sentiment. Feature 'class label' was assigned to the main tweet only and the feature 'bystander role label' was assigned to the reply tweet only. Thus, the dataset had sixteen features for main tweets and fifteen features for reply tweets. (see Table III).

TABLE III. FEATURES OF ORIGINAL DATASET CYBY23

General	Perspective API	TextBlob	Main Tweet	Reply Tweet
tweet_id	Insult	polarity	class label	bystander role label
reply_id	Threat	subjectivity		
text	Identity _Attack	sentiment		
created _at	Profanity			
favorite _count	Toxicity			
retweet _count	Severe _Toxicity			

TABLE IV. FEATURES OF PRE-PROCESSED DATASET

General	Perspective API (Main Tweet)	Perspective API (Reply Tweet)	TextBlob (Main Tweet)	TextBlob (Reply Tweet)	Main Tweet	Reply Tweet
favorite _count	Insult _main	Insult	Polarity _main	Polarity	Class la-bel	Bystander role label
favorite _count _main	Threat _main	Threat	subjectivity _main	Subjectivity		
retweet _count	Identity _Attack _main	Identity _Attack	Sentiment _main	sentiment		
retweet _count _main	Profanity	Profanity				
	Toxicity _main	Toxicity				
	Severe _Toxicity _main	Severe _Toxicity				

B. Data Preprocessing

Certain pre-processing steps were applied to the CYBY23 dataset [12] before running the machine-learning models. Those are listed below:

1) The feature 'bystander role label' had four string values, namely, "This person agrees with the main post (instigator)", "This person disagrees with the main post (defender)", "This person is not taking any sides (impartial)" and "This person posted unrelated

replies (Other)". These string values were converted to numeric values between 0 to 3.

2) To study the effect of the main tweet on the reply tweets, the features of the main tweet were concatenated with the features of the reply tweet, and a new dataset was created. The new dataset had seven general features, six toxicity-related features of the reply tweet and main tweet, three sentiment-related features of the reply tweet and main tweet, feature 'class label' of the main tweet, and feature 'bystander role label' of the reply tweet. Thus, the new dataset had 28 features. Names of main tweet features were suffixed with _main. Since main tweets were concatenated column-wise with reply tweet, so the number of total tweets reduced from 639 to 524.
3) The features tweet_id, reply_id, and created_at were removed as they were not required for the models. So new dataset had 25 features for all the tweets.
4) The feature 'text' was removed from the dataset, because toxicity features using Perspective API and sentiments features using TextBlob had already been computed from the 'text' feature. Thus, the new dataset had 24 features for all the tweets.

After pre-processing, we got the dataset having 524 tweets and 24 features for each tweet (see Table IV). Out of these 24 features, the feature 'bystander role label' was used as the target feature for all machine learning models.

C. Model Development

In this work, we deployed different machine learning models [19] using Pycaret library. A brief description of each of the models is given below:

- AdaBoost Classifier (ADA): Adaptive Boosting Classifier is an ensemble classifier, that benefits from training several weak classifiers and then combining the result, with more weightage given to the classifier that gives more accuracy.
- Decision Tree Classifier (DT): A flowchart-like tree structure where each internal node denotes a test on an attribute, each branch represents the outcome of a test, and each leaf node holds a class label.
- Extra Trees Classifier (et): An ensemble machine learning method based on decision trees. The dataset sampling for each tree is done randomly, without replacement. The features subset is also assigned randomly to each tree.
- Gradient Boosting Classifier (GBC): This classifier is an additive model of decision trees and is often employed for both regression and classification tasks.
- K Neighbors Classifier(KNN): A learning method that uses the nearest neighbors to classify a data point.
- Linear Discriminant Analysis (LDA): A method used to find a linear combination of features that best separates two or more classes in a dataset.
- Light Gradient Boosting Machine (LGBM) & Extreme Gradient Boosting (EGB): Both are gradient boosting frameworks that use tree-based learning algorithms. They are recognized for their efficiency and predictive accuracy.
- Logistic Regression (LR): A foundational statistical method to model the probability of a certain class or event based on one or multiple predictor features.
- Naive Bayes (NB): A probabilistic classifier based on applying Bayes' theorem, it assumes independence between features.
- Random Forest Classifier(rf): An ensemble learning method that uses decision trees. Each decision tree comprises of dataset drawn by bootstrap sampling. The 'majority voting' is used to make final prediction.
- Ridge Classifier (RC): A classification algorithm that employs L2 regularization. It can help prevent overfitting and often delivers better performance in scenarios with multicollinearity.
- SVM - Linear Kernel(SVM): A learning method that finds a hyperplane to separate the two classes such that it maximizes predictive accuracy while avoiding over-fitting.

D. Model Validation

The proposed model was validated using various feature selection techniques:

1) Experimenting on various types of features (Toxicity Based, Sentiment Based)
2) Removal of Highly correlated features
3) Removal of less significant features
4) Hypertuning the parameters of machine learning models

Model efficiency was analyzed after applying each of the techniques mentioned above.

E. Model Evaluation

1) Use of the Tool: We used Pycaret Python library which speeds up the process of experiments related to machine learning and empowers us to run multiple ML models simultaneously. It also helps in hypertuning the parameters of the models which gives us the best performance.
2) Evaluation Metrics: Four metrics, accuracy, precision, recall, and F1 score are used to evaluate the models. Using a wide range of evaluation metrics caters to various aspects of prediction quality.
3) Cross-Validation: We applied K-Fold cross-validation. This method partitioned the training data into 'K' subsets, training on 'K-1' of them and validating on the remaining subset. This process was iteratively executed until each subset had been used for validation, offering a robust average performance metric.
4) Various Train-Test Split: Various splits for training and test sets were used to validate the model.

IV. Experimental Results and Analysis

This section presents the experimental setup, their results, and analysis.

A. Platforms Used

We used Python using Jupyter Notebook and Google collaboratory for running the experiments. Pycaret library was used to run the machine learning models. Plotting of graphs was done using Matplotlib and Pandas library.

B. Dataset

A pre-processed dataset (see Table IV), having 524 tweets and 24 features for each tweet, was used in further experiments.

1) Handling Imbalance of Dataset: The class distribution of the dataset having 524 tweets is shown in Fig. 2 (a). High imbalance can be observed in the number of instances of unique values of the target feature 'bystander role label'. Imbalance can be handled by undersampling or oversampling the minority class. However, undersampling has the chance of losing important information. So, we used an oversampling technique, Synthetic Minority Oversampling Technique (SMOTE) [18] to handle the imbalance. SMOTE generates synthetic samples for the minority class and creates a balanced dataset. Fig. 2 (b) depicts the balanced dataset with 912 data points.

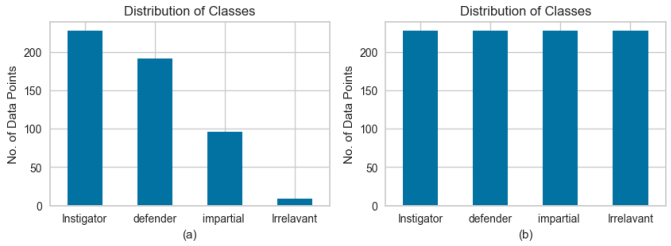

Fig. 2. (a) Class distribution (b) Class distribution after resampling (SMOTE).

C. Model Deployment using Various Feature Selection Techniques

We experimented with different feature selection techniques on various machine learning models. Pycaret was used to run all the models. The models were evaluated using accuracy, precision, recall, and F1 score metrics. The results of running all machine learning models using Pycaret are shown in Table V. The experiments and their results are mentioned below:

- Case 1: Initially we run the experiments using only the toxicity features derived from Perspective API. Random Forest Classifier(rf), Gradient Boosting Classifier (gbc), Light Gradient Boosting Machine (lightgbm), and Extra Trees Classifier (et) performed best, each with accuracy as well as F1 score of 72% (approx).
- Case 2: Further, the experiments were run on Sentiments features derived from TextBlob. Approximately 70% accuracy, and 70% F1 score were achieved using Random Forest Classifier(rf), Gradient Boosting Classifier (gbc), Light Gradient Boosting Machine (lightgbm) and Extra Trees Classifier (et) classifier (see Table V).
- Case 3: Next, we experimented with both the toxicity features (mentioned in case 1) and sentiments features (mentioned in case 2). With this feature set, accuracy as well as F1 score of approx. 75% was achieved with all the four classifiers mentioned in Case 1 and Case 2. Thus, indicating that instead of using only Toxicity or Sentiment features, results are better when both are used.
- Case 4: From the feature set mentioned in case 3, we computed the correlation coefficient among features (see Fig. 3). We found that the feature Severe_Toxicity_main is highly correlated to Toxicity_main. Similarly, the features Profanity and Toxicity, favorite_count_main and retweetcount_main, Toxicity and Insult, Severe_Toxicity and Profanity, Toxicity_main and Insult_main are highly correlated. Thus, we removed the features, 'Severe_Toxicity_main', 'Profanity', 'Toxicity', 'favorite_count_main', 'Toxicity_main, and were left with 19 features.
  After removing the correlated features, the highest accuracy of 76% was achieved. Again, the same four classifiers, rf, gbc, lightgbm, and et, performed best.

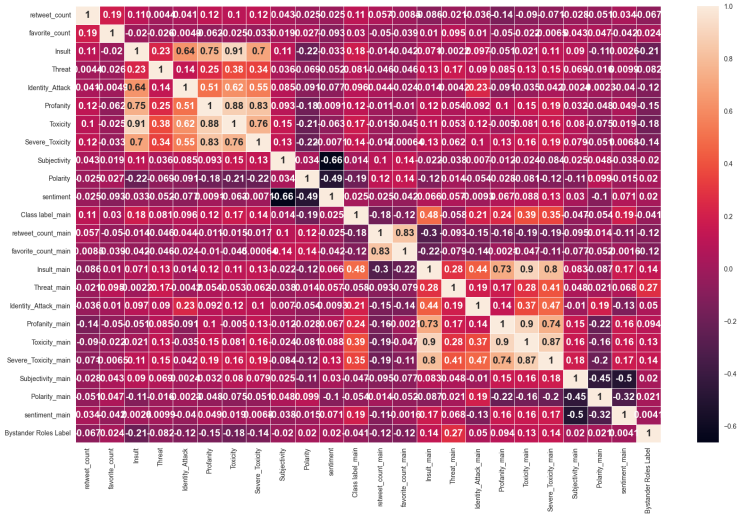

Fig. 3. Heatmap showing correlation among features.

- Case 5: Next, we experimented with finding the importance of the features mentioned in case 3. Some feature ClassLabel_main, sentiment, sentiment_main, and retweet_count were ruled out (see Fig. 4) because of their low importance.
  After removing the less important feature, we checked the efficiency of our models (see Table V). Random Forest Classifier(rf) performed best with 76% accuracy and 78% F1 score.
- Case 6: Further, we chose a feature set that was formed after removing the highly correlated features as well

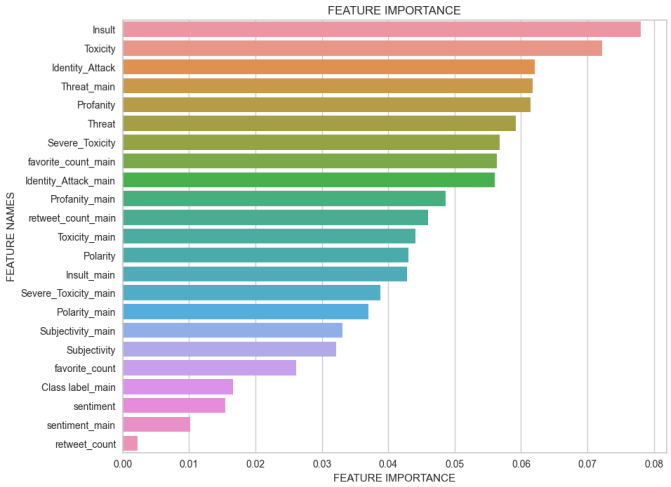

Fig. 4. Feature importance.

as the less important features from the features given in Case 3. Running all the machine learning models using Pycaret gave the results mentioned in Table V. We observe that Random Forest Classifier(rf) again performed best with 77.6% accuracy and 79.8% F1 score.

Fig. 5 compares the accuracy of all the classifier models for each of the feature set case discussed above.

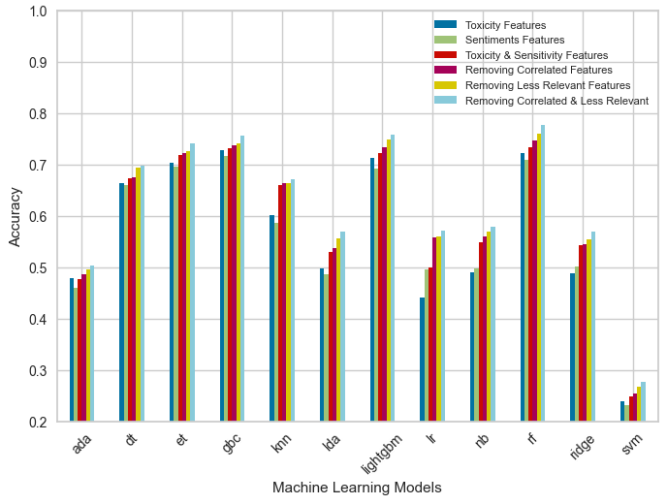

Fig. 5. Comparison of accuracy for different models.

For each of the feature set case discussed above, Fig. 6 compares the accuracy achieved by the different classifier models. Here, results from only those classifier are plotted, which achieved more than 50% accuracy.

As is discussed above for all the six cases and is evident from Fig. 5 and Fig. 6, Random Forest Classifier(rf) performs best for most of cases. Also, the best result is achieved by the feature set formed by including both the Toxicity and Sentiments features and by removing both the least significant features and the highly correlated features.

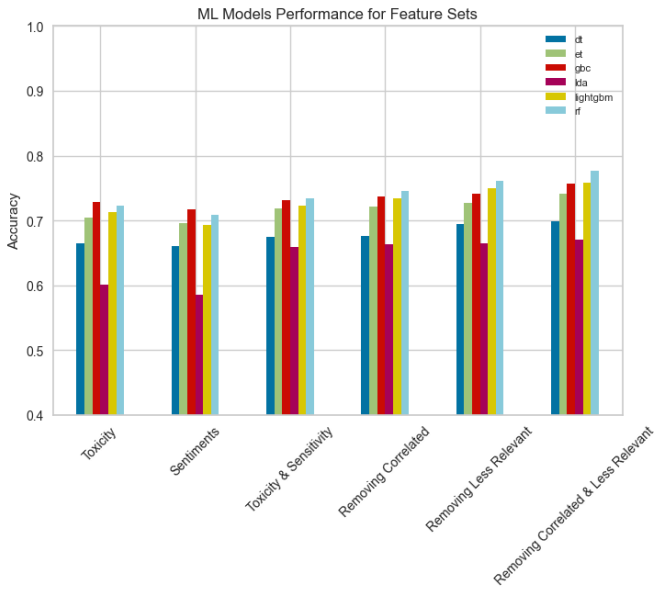

Fig. 6. Comparison of accuracy for different feature sets.

D. Feature Importance

Before going for further experimentation, we would like to give some observations related to the importance of features as depicted in Fig. 4.

1) We found that the features ClassLabel_main, sentiment, sentiment_main, and retweet_count have very less importance as compared to other features (see Fig. 4). This indicates that the level of aggression of the whole thread denoted by ClassLabel_main has little impact on the model performance. The sentiment of the reply tweet and the sentiment of the main tweet has very little role to play along with the number of retweets indicated by retweet_count.
2) Features Insult and Toxicity have the highest importance. One of them can be considered an important feature since they are highly correlated.
3) Feature Threat of the main tweet and reply tweet is almost equally important.
4) Features Identity_Attack, Profanity, Insult, Toxicity, Severe_Toxicity, Polarity, Sentiment of the main tweet have low importance as compared to the corresponding features of the reply tweet except for the Threat and Subjectivity feature.
5) Comparing the set of features based on Perspective API and TextBlob[4], we can observe that features based on Perspective API have more importance.

Summarizing the observations, the top features among all are Toxicity, Identity_Attack, Threat_main, Profanity, and Threat.

E. Different Train-test Split

Then we experimented with different train-test splits for judging the performance of Random Forest classifier. The data

[4]https://textblob.readthedocs.io/en/dev/

TABLE V. EVALUATION METRICS FOR DIFFERENT COMBINATIONS

Model	Accuracy	Recall	Precision	F1-Score
Ada Boost Classifier				
Toxicity Features Only	0.4797	0.4797	0.5374	0.4875
Sentiment Features Only	0.4608	0.5408	0.5489	0.5292
Toxicity & Sentiment Features	0.4766	0.4766	0.5430	0.4838
Removing Correlated Features	0.4872	0.5172	0.56	0.5222
Removing Less Relevant Features	0.4964	0.4764	0.5243	0.4791
Removing Correlated and Less Relevant	0.5031	0.5031	0.5606	0.51
Decision Tree Classifier				
Toxicity Features Only	0.6644	0.6444	0.6435	0.6392
Sentiment Features Only	0.6600	0.6600	0.6609	0.6566
Toxicity & Sentiment Features	0.6740	0.6740	0.6746	0.6693
Removing Correlated Features	0.6756	0.6756	0.6763	0.6725
Removing Less Relevant Features	0.6944	0.6944	0.6988	0.6909
Removing Correlated and Less Relevant	0.6982	0.6912	0.7011	0.6885
Extra Trees Classifier				
Toxicity Features Only	0.7045	0.7445	0.7476	0.7436
Sentiment Features Only	0.6954	0.7054	0.7150	0.7012
Toxicity & Sentiment Features	0.7186	0.7634	0.7662	0.7611
Removing Correlated Features	0.7217	0.7617	0.7665	0.759
Removing Less Relevant Features	0.7266	0.7666	0.7713	0.7641
Removing Correlated and Less Relevant	0.7418	0.7618	0.7668	0.7605
Gradient Boosting Classifier				
Toxicity Features Only	0.7290	0.7290	0.7383	0.7287
Sentiment Features Only	0.7178	0.7178	0.7236	0.7155
Toxicity & Sentiment Features	0.7315	0.7571	0.7590	0.7552
Removing Correlated Features	0.7373	0.7273	0.7278	0.7236
Removing Less Relevant Features	0.7415	0.7415	0.7445	0.7401
Removing Correlated and Less Relevant	0.7563	0.7163	0.7246	0.714
K Neighbors Classifier				
Toxicity Features Only	0.6008	0.6708	0.6976	0.6722
Sentiment Features Only	0.5856	0.6456	0.6644	0.6414
Toxicity & Sentiment Features	0.6597	0.6597	0.6765	0.6596
Removing Correlated Features	0.6633	0.6333	0.6393	0.6282
Removing Less Relevant Features	0.6648	0.6348	0.6516	0.636
Removing Correlated and Less Relevant	0.6706	0.6206	0.613	0.6128
Linear Discriminant Analysis				
Toxicity Features Only	0.4984	0.4984	0.5117	0.4926
Sentiment Features Only	0.4862	0.5062	0.4799	0.4773
Toxicity & Sentiment Features	0.5301	0.5501	0.5560	0.5435
Removing Correlated Features	0.5376	0.5376	0.5439	0.5329
Removing Less Relevant Features	0.5556	0.5156	0.5255	0.5127
Removing Correlated and Less Relevant	0.5687	0.4987	0.517	0.4991
Light Gradient Boosting Machine				
Toxicity Features Only	0.7136	0.7336	0.7358	0.7311
Sentiment Features Only	0.6931	0.7131	0.7187	0.7090
Toxicity & Sentiment Features	0.7233	0.7633	0.7648	0.7617
Removing Correlated Features	0.7337	0.7337	0.7368	0.73
Removing Less Relevant Features	0.7495	0.7695	0.7726	0.7676
Removing Correlated and Less Relevant	0.7587	0.7587	0.7678	0.757
Logistic Regression				
Toxicity Features Only	0.4404	0.4404	0.4339	0.4219
Sentiment Features Only	0.4953	0.4953	0.4662	0.4593
Toxicity & Sentiment Features	0.5000	0.5000	0.4883	0.4815
Removing Correlated Features	0.5579	0.5579	0.5389	0.5409
Removing Less Relevant Features	0.5601	0.4701	0.4741	0.4666
Removing Correlated and Less Relevant	0.5722	0.5222	0.52	0.5099
Naive Bayes				
Toxicity Features Only	0.4907	0.4907	0.5382	0.4473
Sentiment Features Only	0.4984	0.4984	0.5520	0.4436
Toxicity & Sentiment Features	0.5485	0.5485	0.5881	0.5115
Removing Correlated Features	0.5593	0.5393	0.5775	0.4991
Removing Less Relevant Features	0.5691	0.4591	0.471	0.4204
Removing Correlated and Less Relevant	0.5791	0.4891	0.5066	0.4399
Random Forest Classifier				
Toxicity Features Only	0.7226	0.7226	0.7233	0.7200
Sentiment Features Only	0.7085	0.7085	0.7111	0.7036
Toxicity & Sentiment Features	0.7346	0.7346	0.7514	0.7442
Removing Correlated Features	0.7462	0.7462	0.7773	0.7661
Removing Less Relevant Features	0.7606	0.7606	0.7869	0.7778
Removing Correlated and Less Relevant	0.7766	0.7666	0.7987	0.7938
Ridge Classifier				
Toxicity Features Only	0.4890	0.4890	0.4739	0.4686
Sentiment Features Only	0.5016	0.5016	0.4623	0.4531
Toxicity & Sentiment Features	0.5439	0.5439	0.5182	0.5155
Removing Correlated Features	0.5455	0.5455	0.5264	0.521
Removing Less Relevant Features	0.5547	0.5047	0.4969	0.4889
Removing Correlated and Less Relevant	0.5697	0.5097	0.5067	0.4952
SVM Linear Kernel				
Toxicity Features Only	0.2397	0.2397	0.1374	0.1305
Sentiment Features Only	0.2321	0.2321	0.1438	0.1190
Toxicity & Sentiment Features	0.2492	0.2492	0.2285	0.1450
Removing Correlated Features	0.2541	0.2541	0.1733	0.1568
Removing Less Relevant Features	0.2664	0.2664	0.2388	0.1434
Removing Correlated and Less Relevant Features	0.2765	0.262	0.2297	0.1971

was split in multiple split percentages, starting from 60% till 95% with a window of 5%. Fig. 7 summarizes the results of running the Random forest classifier when the training set is split from 60% till 95% with a window of 5%. We observe that the accuracy increases with the increase in the size of the training set but almost stabilizes when it reaches 85%. Similar is the case with F1 score. Hence, best accuracy of 78.83% and F1 score of 79.45% is reported to be achieved at 85% training set and 15% test set (see Table VI).

Fig. 7. Model performance on different train-test split.

TABLE VI. MODEL PERFORMANCE FOR VARIOUS TRAIN-TEST SPLITS

Train-Test Split	Accuracy	Precision	Recall	F1 Score
60-40	0.7150	0.7241	0.7050	0.7119
65-35	0.7281	0.7509	0.7081	0.7277
70-30	0.7189	0.7506	0.7010	0.7204
75-25	0.7280	0.7637	0.7080	0.7308
80-20	0.7486	0.7846	0.7286	0.7520
85-15	0.7883	0.8179	0.7483	0.7945
90-10	0.7883	0.8126	0.7434	0.7945

V. CONCLUSION AND FUTURE WORK

In this paper, a machine learning model for automatic labeling of Bystanders detection has been proposed. Initially, Pycaret was used to find the best model using the features mentioned in the CYBY23 dataset [12]. Later, various feature selection techniques have been used to increase the efficiency of the model. The proposed model has been validated by using different train-test splits. The results of various combinations has been discussed in length. Finally, the Random Forest classifier with a training set of 85% and 15% has been chosen as the best model for Bystanders detection. Further, the Importance of various features from the given dataset has been discussed. Despite the best efforts of applying machine learning techniques for the given dataset CYBY23 [12], the authors feel that the small size of the dataset hinders the research in this area. The achieved results will be more promising on a larger dataset.

The research work in the future may be directed toward increasing the dataset size and finding a more efficient model for automatic labeling. The dataset size can be increased by extending the work to other social media posts. Various ways of finding the sentiments can be used using Natural Language Processing techniques. Deep learning models can be experimented with for the deployment of an efficient model for automatic labeling. The mentioned dataset can be regarded as a multi-label dataset with two class labels, namely aggression level, and bystanders role and further experiments can be performed in that direction.

ACKNOWLEDGMENT

We are thankful for the facilities given in our academic institutes, SSCBS and P.G.D.A.V. college of the University of Delhi.

REFERENCES

[1] T. Mahlangu, C. Tu, and O. Pius, "A review of automated detection methods for cyberbullying," in *International Conference on Intelligent and Innovative Computing Applications (ICONIC)*. IEEE, 12 2018, pp. 1–5.

[2] H. Kallmen and M. Hallgren, "Bullying at school and mental health problems among adolescents: a repeated cross-sectional study." *Child Adolesc Psychiatry Ment Health*, vol. 74, 2021. [Online]. Available: https://doi.org/10.1186/s13034-021-00425-y

[3] S. Salawu, Y. He, and J. Lumsden, "Approaches to automated detection of cyberbullying: A survey," *IEEE Transactions on Affective Computing*, vol. 11, no. 1, pp. 3–24, Mar. 2020.

[4] C. Van Hee, G. Jacobs, C. Emmery, B. Desmet, E. Lefever, B. Verhoeven, G. De Pauw, W. Daelemans, and V. Hoste, "Automatic detection of cyberbullying in social media text," *PLOS ONE*, vol. 13, no. 10, p. e0203794, Oct. 2018.

[5] P. K. Smith, J. Mahdavi, M. Carvalho, S. Fisher, S. Russell, and N. Tippett, "Cyberbullying: its nature and impact in secondary school pupils," *Journal of Child Psychology and Psychiatry*, vol. 49, no. 4, pp. 376–385, 2008. [Online]. Available: https://acamh.onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-7610.2007.01846.x

[6] C. Van Hee, E. Lefever, B. Verhoeven, J. Mennes, B. Desmet, G. De Pauw, W. Daelemans, and V. Hoste, "Automatic detection and prevention of cyberbullying," in *International Conference on Human and Social Analytics, Proceedings*, P. Lorenz and C. Bourret, Eds. IARIA, 10 2015, pp. 13–18.

[7] M. Tsvetkova and M. Macy, "The social contagion of antisocial behavior," *Sociological Science*, vol. 2, pp. 36–49, 02 2015.

[8] E. J. Villota and S. G. Yoo, "An experiment of influences of facebook posts in other users," in *2018 International Conference on eDemocracy and eGovernment (ICEDEG)*, Ambato, Ecuador, 2018, pp. 83–88.

[9] K. Yokotani and M. Takano, "Social contagion of cyberbullying via online perpetrator and victim networks," *Computers in Human Behavior*, vol. 119, p. 106719, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0747563221000418

[10] G. Rathnayake, T. Atapattu, M. Herath, G. Zhang, and K. Falkner, "Enhancing the identification of cyberbullying through participant roles," in *Proceedings of the Fourth Workshop on Online Abuse and Harms*. Association for Computational Linguistics, 11 2020, pp. 89–94. [Online]. Available: https://aclanthology.org/2020.alw-1.11

[11] Jacobs, Gilles and Van Hee, Cynthia and Hoste, Veronique, "Automatic classification of participant roles in cyberbullying: can we detect victims, bullies, and bystanders in social media text?" *Natural Language Engineering*, vol. 28, no. 2, pp. 141–166, 2022. [Online]. Available: http://doi.org/10.1017/s135132492000056X

[12] H. Alfurayj, N. S. Yee, and S. L. Lutfi, "Cyberbullying bystander dataset 2023," 2023. [Online]. Available: https://www.kaggle.com/dsv/6486152

[13] H. S. Alfurayj, S. L. Lutfi, and N. S. Yee, "Bystanders unveiled: Introducing a comprehensive cyberbullying corpus with bystander information," *TENCON 2023 - 2023 IEEE Region 10 Conference (TENCON)*, pp. 1012–1017, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:265354251

[14] H. S. Alfurayj and S. L. Lutfi, "Exploring bystanders' roles in labeled cyberbullying threads on twitter: A preliminary analysis," *TENCON 2023 - 2023 IEEE Region 10 Conference (TENCON)*, 2023.

[15] R. C. J.I. Sheeba, S. Pradeep Devaneyan, "Identification and classification of cyberbully incidents using bystander intervention model," *International Journal of Recent Technology and Engineering*, vol. 8, no. 2S4, p. 1–6, Aug. 2019. [Online]. Available: http://dx.doi.org/10.35940/ijrte.B1001.0782S419

[16] A. Aleksandric, M. Singhal, A. Groggel, and S. Nilizadeh, "Understanding the bystander effect on toxic twitter conversations," *ArXiv*, vol. abs/2211.10764, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:253734314

[17] M. Obermaier, N. Fawzi, and T. Koch, "Bystanding or standing by? how the number of bystanders affects the intention to intervene in cyberbullying," *New Media & Society*, vol. 18, no. 8, pp. 1491–1507, 2016.

[18] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *J. Artif. Intell. Res. (JAIR)*, vol. 16, pp. 321–357, 06 2002.

[19] I. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *SN Computer Science*, vol. 2, 03 2021.

[20] Van Hee, Cynthia and Verhoeven, Ben and Lefever, Els and De Pauw, Guy and Daelemans, Walter and Hoste, Veronique, "Guidelines for the fine-grained analysis of cyberbullying, version 1.0," 2015.