International Journal of Advancement in Electronics and Computer Engineering (IJAECE)
Volume 13, Issue 6, June. 2024, pp.139-145, ISSN 2278 -1412

COMPARATIVE EVALUATION OF MACHINE LEARNING CLASSIFIERS FOR CYBERBULLYING DETECTION

[1] Nidhi koyale, [2]Dr. Pushparaj Singh Chauhan
[1]M. Tech. Scholar, CSE SISTECH Bhopal, nidhikoyle22@gmail.com, India
[2]Prof. & Head of Dept., CSE SISTECH Bhopal,hodcybersecurity@sistec.ac.in, India

Abstract: - *In this paper evaluated the performance of several machine learning classifiers in detecting cyberbullying. The models tested include BaggingClassifier, SGDClassifier, LogisticRegression, DecisionTreeClassifier, RandomForestClassifier, LinearSVC, AdaBoostClassifier, MultinomialNB, and KNeighborsClassifier. The evaluation metrics considered were Accuracy, Precision, Recall, and F1 Score. Among the tested models, the BaggingClassifier emerged as the top performer with an accuracy of 0.928, a precision of 0.964, a recall of 0.925, and an F1 score of 0.944, indicating its high effectiveness and balance between precision and recall. The SGDClassifier followed closely, achieving an accuracy of 0.927, a precision of 0.958, a recall of 0.930, and an F1 score of 0.944, demonstrating excellent performance as well. The LogisticRegression model also showed strong results with an accuracy of 0.926, a precision of 0.964, a recall of 0.922, and an F1 score of 0.943. DecisionTreeClassifier and RandomForestClassifier achieved slightly lower accuracies of 0.923 and 0.919, respectively, but maintained strong precision and recall. LinearSVC had an accuracy of 0.917, while AdaBoostClassifier, MultinomialNB, and KNeighborsClassifier showed lower accuracies of 0.908, 0.893, and 0.858, respectively, indicating their relative ineffectiveness for this task. The results suggest that BaggingClassifier and SGDClassifier are highly reliable choices for cyberbullying detection, with LogisticRegression also being a strong contender.*

Keywords: - Cyberbullying Detection, Machine Learning Classifiers, Comparative Evaluation, Cyberbullying Analysis, Classification Algorithms, Text Analysis, Social Media Monitoring

I. INTRODUCTION

Social networking sites have become integral to modern life, serving as hubs for entertainment, networking, and communication for billions of people worldwide. These platforms have revolutionized the way we interact, offering new dimensions to communication by connecting individuals across the globe. However, the widespread use of social media has also given rise to serious issues such as cyberbullying. This form of harassment, particularly prevalent among teenagers, involves the use of electronic technologies to deliberately and repeatedly intimidate or threaten others. Common platforms for such abuse include Twitter, Facebook, and email, where bullies exploit these services to target victims through various means such as fake identities, embarrassing posts, and threats.

The consequences of cyberbullying are severe, often leading to significant psychological distress and, in some tragic cases, even death. As a result, there is a pressing need for effective solutions to detect and prevent cyberbullying. Leveraging machine learning approaches offers a promising avenue for addressing this problem from a fresh perspective. By understanding the mechanisms and impacts of cyberbullying, we can develop more sophisticated methods to combat it and create safer online environments. As social networking sites increasingly take steps to mitigate such abuses, ongoing research and technological advancements will be crucial in the fight against cyberbullying.

One of the most alarming aspects of cyberbullying is its potential to escalate quickly. A single harmful post or message can be shared and viewed by countless people within minutes, magnifying the victim's humiliation and distress. The viral nature of social media means that victims may find themselves subjected to widespread ridicule and ostracism, exacerbating their sense of isolation.

The consequences of cyberbullying can be devastating. Victims often experience a range of negative emotions, including fear, anger, and helplessness. The relentless nature of cyberbullying can lead to long-term psychological issues, such as chronic anxiety, depression, and post-traumatic stress disorder (PTSD). In extreme cases, the impact of cyberbullying has been linked to suicidal behavior, highlighting the urgent need for effective interventions.

Figure 1 Cyber Bullying

Addressing cyberbullying requires a comprehensive approach. Education is a critical component, as raising

DOI Member 109.49.410

awareness about the issue and teaching digital citizenship can empower individuals to recognize and respond to cyberbullying. Schools, parents, and communities must collaborate to provide support and resources for victims while promoting a culture of empathy and respect. Technological solutions also play a crucial role in combating cyberbullying. Machine learning and artificial intelligence can be employed to monitor online interactions and detect harmful behavior. These technologies can analyze text, images, and user activity to identify patterns indicative of cyberbullying. By flagging and addressing these behaviors promptly, it is possible to prevent escalation and provide timely support to victims.

Moreover, social media platforms must continue to enhance their policies and tools to combat cyberbullying effectively. This includes improving reporting mechanisms, enforcing stricter penalties for offenders, and offering resources for users to protect themselves. Collaboration with law enforcement and mental health professionals is also essential to provide comprehensive support for victims and hold perpetrators accountable.

II. PROPOSED METHOD

The proposed method leverages advanced machine learning techniques to develop an automated system capable of detecting cyberbullying on social media platforms. This method comprises several key stages, including data collection, preprocessing, feature extraction, model training, and deployment. Each stage is designed to address specific challenges associated with detecting cyberbullying in real-time while ensuring high accuracy and low false-positive rates.

Dataset Description

Social media platforms have become the most prominent medium for spreading hate speech, primarily through hateful textual content. This extensive dataset is is used to design models to detect hate speech on social media, incorporating current trends in online communication, such as the use of emoticons, emojis, hashtags, slang, and contractions. The dataset includes hate speech sentences in English, categorized into two classes: one representing hateful content and the other representing non-hateful content.

This dataset falls under the subject of Natural Language Processing (NLP), specifically focusing on a curated collection of text data comprising emojis, emoticons, and contractions. The data is annotated, analyzed, and filtered, with each text sample classified as either hateful or non-hateful. The data article titled "A curated dataset for hate speech detection on social media text" can be accessed at [Mendeley Data](https://data.mendeley.com/datasets/9sxpkmm8xn/)

Algorithm

1. Prepare and Load Dataset
   a. Load the dataset from the specified filepath into a data structure (e.g., a table or dataframe).

2. Data Pre-Processing
   a. Handle missing values and duplicate values:
      i. Remove duplicate entries from the dataset.
      ii. Remove entries with missing values from the dataset.
   b. Convert all text data to lowercase.
   c. Remove non-word characters from the text data.
   d. Tokenization:
      i. Split the text data into individual words (tokens).
   e. Removing Stopwords:
      i. Create a list of common stopwords.
      ii. Remove stopwords from the tokenized text data.
   f. Convert tokenized text back to a string format for further processing.

3. Train-Test Split (80:20)
   a. Split the dataset into training and testing sets with an 80:20 ratio.
      i. The training set will be used to train the machine learning model.
      ii. The testing set will be used to evaluate the model's performance.

4. Apply Machine Learning for Training and Detection of Cyberbullying
   a. Vectorize the text data using a suitable vectorization method (e.g., TF-IDF).
   b. Train a machine learning model using the training data.
   c. Use the trained model to make predictions on the test data.

5. Calculate Accuracy and Result
   a. Compare the model's predictions with the actual labels in the test data.
   b. Calculate the accuracy of the model based on the comparison.
   c. Output the accuracy and other relevant results.

6. Main Function to Execute the Workflow
   a. Load the dataset.
   b. Preprocess the data.
   c. Split the data into training and testing sets.
   d. Train the machine learning model.
   e. Make predictions using the trained model.
   f. Calculate and display the accuracy of the model.

This dataset is valuable for training machine learning models to identify hate speech on social media. It captures current social media trends and modern communication styles, making it useful for developing systems to automatically filter out hateful content. Social media managers, administrators, and companies can utilize this dataset to categorize text as hateful or

non-hateful. Deep Learning (DL) and Natural Language Processing (NLP) practitioners can benefit from using this dataset to detect hateful speech through various techniques. The text samples are labeled with "0" for non-hateful and "1" for hateful, making it a useful benchmark for hate speech detection.

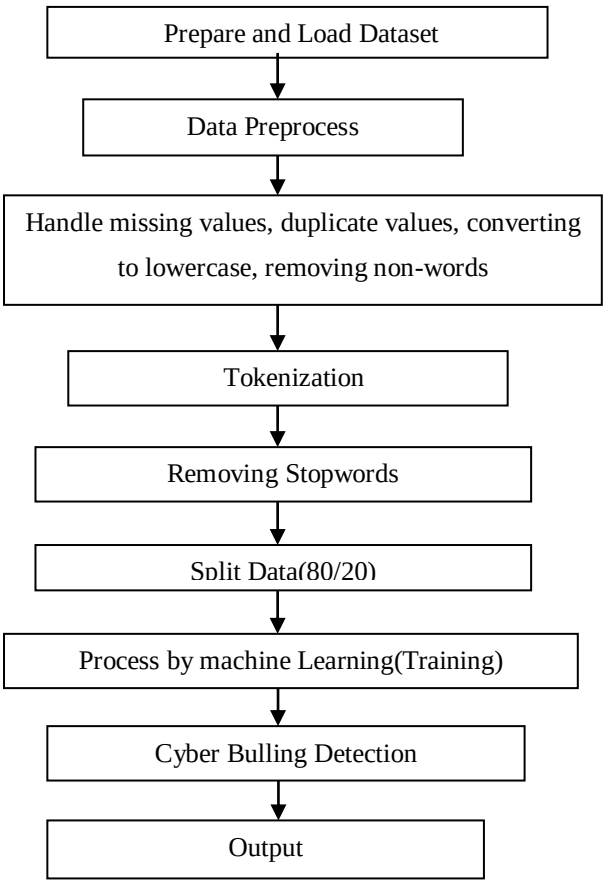

Figure 2 Proposed Flow

For feature extraction, the `CountVectorizer` method from scikit-learn is instantiated. This method converts the text data into a matrix of token counts, which is a common approach in text classification tasks. By specifying `stop_words='english'` and `lowercase=True`, the vectorizer ensures that the text is further normalized and stopwords are excluded from the feature set.

Tokenization:
The tokenizer is used to tokenize the input texts. Each text is converted into a sequence of tokens, with special tokens [CLS] at the beginning and [SEP] at the end to signify the start and end of the input sequence, respectively.

$$tokens = tokenizer.tokenize(text)$$
$$input\_tokens = ['[CLS]'] + tokens[:510] + ['[SEP]']$$

The training data is fitted and transformed into a matrix of token counts, which creates a numerical representation of the text suitable for machine learning algorithms. The testing data is then transformed using the same vectorizer, ensuring that both training and testing data are represented consistently.

III. RESULT

This dataset is valuable for training machine learning models to identify hate speech on social media. It captures current social media trends and modern communication styles, making it useful for developing systems to automatically filter out hateful content. Social media managers, administrators, and companies can utilize this dataset to categorize text as hateful or non-hateful. Deep Learning (DL) and Natural Language Processing (NLP) practitioners can benefit from using this dataset to detect hateful speech through various techniques. The text samples are labeled with "0" for non-hateful and "1" for hateful, making it a useful benchmark for hate speech detection.

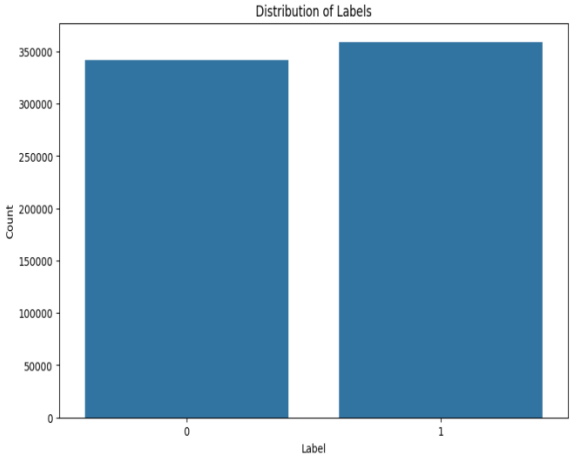

Figure 3 Data Distribution

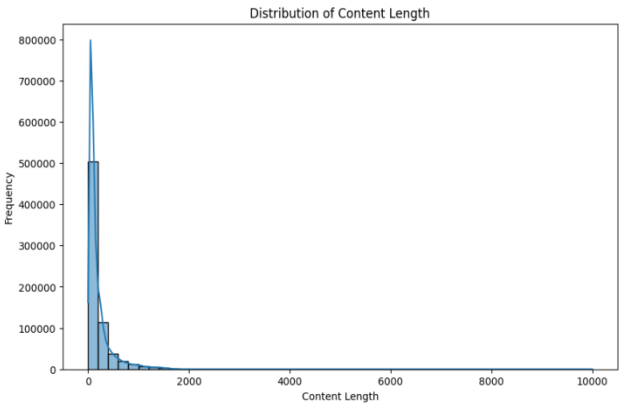

Figure 4. Content Length Distribution

Data Preprocessing

We begin by loading data, preprocesing, and finally splitting the dataset for task aimed at detecting hate speech. The first step involves reading the data from a

CSV file into a pandas DataFrame, allowing for easy manipulation and analysis of the dataset.

index	label	full_text	Content Length
0	1	So Drasko just said he was impressed the girls cooked half a chicken.. They cooked a whole one #MKR	100
2	1	Drasko they didn't cook half a bird you idiot #mkr	50
4	1	Hopefully someone cooks Drasko in the next ep of #MKR	53
6	1	of course you were born in serbia...you're as fucked as A Serbian Film #MKR	75
7	1	These girls are the equivalent of the irritating Asian girls a couple years ago. Well done, 7. #MKR	99
8	1	#MKR Lost the plot - where's the big Texan with the elephant sized steaks that they all have for brekkie ?	107
10	1	RT @PhxKen: SIR WINSTON CHURCHHILL: "ISLAM IS A DANGEROUS IN A MAN AS RABIES IN A DOG" http://t.co/kCXgKD70SK	109
11	1	RT @TheRightWingM: Giuliani watched his city attacked & people jump to their deaths. He's entitled to say WTF he wants about the guy shield...	144
12	1	RT @YesYoureRacist: At least you're only a tiny bit racist RT @AnMo95: I'm not racist, but my dick is!	102
13	1	@MisfitInChains @oldgfatherclock @venereveritas13 SANTA JUST "IS" WHITE	71
14	1	RT @Dreamdefenders: Eric Holder from #ferguson: "I understand that mistrust. I am the Attorney General, but I am also a Black Man" http://t...	140
15	1	RT @AntonioFrench: I spent the morning at the Board of Elections getting maps/data to start registering every black person in #Ferguson. ht...	140

Figure 5 Content Classification

In the preprocessing phase, the text data is converted to lowercase to ensure uniformity, which helps in reducing the complexity of the text analysis by treating words like 'Hate' and 'hate' as the same. This conversion is done by transforming the `Content` column of the DataFrame. A set of stopwords is defined using NLTK's English stopwords list. The `clean_text` function is then created to further process the text. This function removes all non-alphanumeric characters using regular expressions, tokenizes the text into individual words, and filters out any stopwords, retaining only meaningful words. The cleaned text is then recombined into a single string.After cleaning, the dataset is split into training and testing sets using the `train_test_split` method from scikit-learn. This method divides the dataset into two subsets: one for training the machine learning model and another for testing its performance. The `random_state` parameter ensures reproducibility of the split. The code then prints the number of rows in the entire dataset, the training set, and the test set to verify the split.

index	Content	Label	Content Length
0	denial of normal the con be asked to comment on tragedies an emotional retard	1	77
1	just by being able to tweet this insufferable bullshit proves trump a nazi you vagina	1	85
2	that is retarded you too cute to be single that is life	1	55
3	thought of a real badass mongol style declaration of war the attackers capture a citizen of the soon to be	1	106
4	afro american basho	1	19
5	yeah retard haha	1	16
6	the ching chong chung stuff	1	27
7	the dead what a slut still warm when she tweeted this it what a slut that vagina her mate obama who sent the fucking lowlife	1	124
8	let your tweets be harmless it will not affect me by the way i am not the faggot one she is n	1	93
9	these latinos who have a problem with immigration enforcement should stay in the shithole	1	89
10	i feel so much secondhand embarrassment when a white person calls me ppi or milo just say spic like you want to and move on	1	123
11	you have got a gorgeous figure what an unfunny twat	1	51
12	what a vile vagina	1	18
13	oh shut up you twat	1	19
14	because i can you fucking retard what does up it up twat chops	1	62
15	this fucking faggot	1	19

Figure 6 Data leveling

The performance of four different machine learning models—Multinomial Naive Bayes (MultinomialNB), Linear Support Vector Classification (LinearSVC), AdaBoost Classifier, and Logistic Regression—has been evaluated across various metrics, including training and prediction times, accuracy, F1 score, precision, and recall for both test and training datasets.

Table:1: Compare Table

Algorithm	Training Time (sec)	Prediction Time (sec)	Accuracy : Test	Accuracy : Train
Multinomial NB	0.26	0.20	0.7956	0.8051
LinearSVC	224.85	0.15	0.8412	0.8886
AdaBoostClassifier	73.91	20.14	0.6923	0.6915
LogisticRegression	18.83	0.10	0.8323	0.8582

Algorithm	F1 Score: Test	F1 Score: Train	Precision: Test	Precision: Train	Recall: Test	Recall: Train
Multinomial NB	0.8202	0.8278	0.7473	0.7556	0.9090	0.9153
LinearSVC	0.8478	0.8919	0.8339	0.8863	0.8622	0.8975
AdaBoostClassifier	0.7513	0.7506	0.6417	0.6404	0.9059	0.9066

MultinomialNB is the fastest in terms of both training and prediction time, taking only 0.26 seconds and 0.20 seconds respectively. It achieves a test accuracy of 0.7956 and a slightly higher training accuracy of 0.8051. The F1 scores are 0.8202 for the test set and 0.8278 for the training set, indicating a small amount of overfitting. The model's precision and recall for the test set are 0.7473 and 0.9090 respectively, suggesting that while it may have a fair number of false positives, it is highly effective at identifying relevant instances.

LinearSVC, while taking significantly longer to train at 224.85 seconds, has a swift prediction time of 0.15 seconds. It achieves the highest test accuracy among the models at 0.8412, with the training accuracy even higher at 0.8886. This disparity points to some degree of overfitting. The F1 score is 0.8478 for the test set and 0.8919 for the training set, further indicating overfitting. Precision and recall for the test set are 0.8339 and 0.8622 respectively, which shows that this model strikes a good balance between precision and recall.

The AdaBoost Classifier has a moderate training time of 73.91 seconds but a significantly longer prediction time of 20.14 seconds. Its test and training accuracies are closely matched at around 0.6923 and 0.6915 respectively, suggesting that the model generalizes well but has lower overall performance compared to the other models. The F1 scores are 0.7513 for the test set and 0.7506 for the training set. Precision for the test set is 0.6417, and recall is high at 0.9059, indicating that while the model is good at identifying relevant

instances, it also has a relatively high rate of false positives.

Logistic Regression balances training and prediction times well, at 18.83 seconds and 0.10 seconds respectively. It achieves a test accuracy of 0.8323 and a training accuracy of 0.8582. The F1 scores are 0.8377 for the test set and 0.8625 for the training set, suggesting some overfitting. Precision and recall for the test set are 0.8320 and 0.8434 respectively, indicating that this model maintains a good balance between identifying relevant instances and limiting false positives.

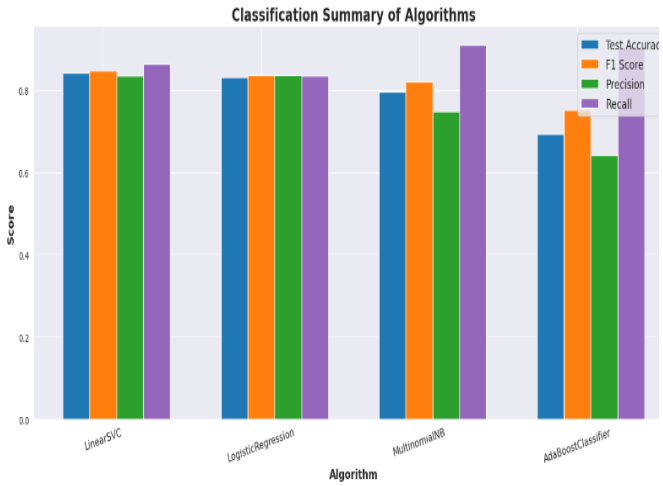

Figure 7: Classification summary of proposed method

IV. CONCLUSION

In this paper evaluated the performance of several machine learning classifiers in detecting cyberbullying. The models tested include Bagging Classifier, SGD Classifier, Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Linear SVC, Ada Boost Classifier, Multinomial NB, and K Neighbors Classifier. The metrics considered for evaluation were Accuracy, Precision, Recall, and F1 Score.

The Bagging Classifier emerged as the top performer with an accuracy of 0.928, a precision of 0.964, a recall of 0.925, and an F1 score of 0.944. This indicates that the Bagging Classifier is highly effective, balancing precision and recall well, and making it a reliable choice for cyberbullying detection. Close behind, the SGD Classifier achieved an accuracy of 0.927, a precision of 0.958, a recall of 0.930, and an F1 score of 0.944, demonstrating excellent performance and making it a strong contender alongside the Bagging Classifier.

The Logistic Regression model also performed well, with an accuracy of 0.926, a precision of 0.964, a recall of 0.922, and an F1 score of 0.943. Decision Tree Classifier and Random Forest Classifier achieved slightly lower accuracies of 0.923 and 0.919 respectively, but still showed strong performance in terms of precision and recall. Linear SVC followed closely with an accuracy of 0.917, while Ada Boost Classifier, Multinomial NB, and K Neighbors Classifier had lower accuracies of 0.908, 0.893, and 0.858 respectively, indicating that they are less effective for this particular task.

.

REFERENCES

[1] Aditya Desai, Shashank Kalaskar, Omkar Kumbhar and Rashmi Dhumal, "Cyber Bullying Detection on Social Media using Machine Learning" International Conference on Automation, Computing and Communication, Volume 40, 2021, DOI: https://doi.org/10.1051/itmconf/20214003038

[2] Abdhullah-Al-Mamun and Shahin Akhter , "Social media bullying detection using machine learning on Bangla text" 2018 10th International Conference on Electrical and Computer Engineering (ICECE), DOI: 10.1109/ICECE.2018.8636797

[3] Vikas S Chavan; Shylaja S S, "Machine learning approach for detection of cyber-aggressive comments by peers on social media network", 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI), DOI: 10.1109/ICACCI.2015.7275970

[4] Elif Varol Altay; Bilal Alatas, "Detection of Cyberbullying in Social Networks Using Machine Learning Methods", 2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), DOI:10.1109/IBIGDELFT.2018.8625321

[5] Mohammed Ali Al-Garadi and Mohammad Rashid Hussain, "Predicting Cyberbullying on Social Media in the Big Data Era Using Machine Learning Algorithms: Review of Literature and Open Challenges", Vol 7, IEEE, DOI: 10.1109/ACCESS.2019.2918354

[6] Rashi Shah, Srushti Aparajit, Riddhi Chopdekar, Rupali Patil, "Machine Learning based Approach for Detection of Cyberbullying Tweets", International Journal of Computer Applications, Volume 175 – No. 37, December 2020

[7] Patxi Galán-García, José Gaviria de la Puerta, Carlos Laorden Gómez, Igor Santos, Pablo García Bringas, "Supervised machine learning for the detection of troll profiles in twitter social network: application to a real case of cyberbullying", Logic Journal of the IGPL, Volume 24, Issue 1,2016,DOI: https://doi.org/10.1093/jigpal/jzv048

[8] Amgad Muneer & Suliman Mohamed Fati , "A Comparative Analysis of Machine Learning Techniques for Cyberbullying Detection on Twitter",

Vol 12 Issue 11, MPDI,2020 , DOI: https://doi.org/10.3390/fi12110187

[9] Batoul Haidar*, Maroun Chamoun, Ahmed Serhrouchni," A Multilingual System for Cyberbullying Detection: Arabic Content Detection using Machine Learning", Advances in Science, Technology and Engineering Systems Journal Vol. 2, No. 6, 275-284 (2017) .
[10] Manuel F. López-Vizcaíno, Francisco J. Nóvoa, Victor Carneiro, Fidel Cacheda," Early detection of cyberbullying on social media networks", Future Generation Computer Systems 118 (2021) 219–229

[1]