# Design And Comparison Of Traditional Cyberbullying Detection Models Using NLP

Megha Manoj†
*Department of Computer Science*
*BITS Pilani, Dubai Campus*
Dubai, United Arab Emirates
f20200016@dubai.bits-pilani.ac.in

Dr. Sujala D. Shetty
*Department of Computer Science*
*BITS Pilani, Dubai Campus*
Dubai, United Arab Emirates
sujala@dubai.bits-pilani.ac.in

***Abstract*— This paper intends to research various techniques used in the detection and prevention of cyberbullying and implement them. Bullying refers to the act of harassing an individual or group to inflict humiliation or superiority through physical violence and cause mental distress or both. With the rise and evolution of technology and the availability of social media as a platform of expression the act of cyberbullying or online bullying has significantly increased. To counter the issue of online bullying various anti-cyberbullying campaigns as well as legal actions have been proposed and launched where one can report such actions as cybercrimes. In an attempt to automate this particular process various technological models have been proposed to classify media content as cyberbullying and the latter. With respect to this advancement a web application will be developed using traditional models to compare the generated labels for text-based cyberbullying and thereby showcase the performance of each model.**

***Keywords*— *NLP, TF-IDF transformation, cyberbullying, SVM, Ensemble learning, GloVe vectorization, Naïve Bayes, XGBoost, LSTM, Bi-LSTM***

## I. Introduction

Cyberbullying is a prominent issue that has become increasingly frequent with the evolution and rise of new internet platforms. According to an article released by broadband research [10], 44% of individuals have claimed that they were bullied in the past 30 days while 73% of students admit they were victims of cyberbullying at some point in their life. While numerous instances of online bullying are identified and classified with ease by humans, the same cannot be said for the automated machine-learning process of detecting such harassment. To tackle and understand the issue of cyberbullying via research, a web application will be built to compare the responses of various traditional machine learning models to gain a better insight into cyberbullying classification via text.

## II. Literature Survey

### *A. Definition of Cyberbullying*

Bullying is referred to as constant aggression towards an individual or group where the assailants intentionally inflict harm on an individual or group [4]. Different types of behaviours can be interpreted as cyber-aggression. These include teasing, demoralizing, offensive, rude, or insulting comments through online social media aimed at individuals or communities in terms of textual cyberbullying [5]. T. Davidson, D. Warmsley, M. Macy, and I. Weber [1] has tried to define what kind of speech may classify as hate speech and thereby establish a boundary separating hate speech from the offensive language in the field of automated hate speech detection. According to him, the definition of such language is quite difficult due to the varied use of language under different instances, such as the use of slur words like "n*gger" depicting African-Americans may differ based on "who" uses the term and "where". While the use of vulgar language is a primary factor of cyberbullying, a feature that distinguishes cyberbullying from other forms of bullying is the lack of a relationship between the bully and the victim. Indicators of cyberbullying include the frequency of certain actions may also fall under it. Hence, cyberbullying may be an elaborate act of online aggression that is repeatedly performed and acts of trolling, and sexual exploitation also fall under it [2]. The fast-paced evolution of online social media inhibits young users from identifying socially acceptable and harmful behaviours when engaging with such platforms. In addition to exposure to this form of existing vulnerability, the speedy nature of resharing information worsens the effects, driving the victims to develop suicidal instincts in severe cases [6]. In countries like the USA, cyberbullying has been recognized as a societal threat which requires exploration to understand how it can occur and devise prevention methods. The challenges in mitigating online bullying starts at how to recognize such aggression and report it to the relevant authorities through automation [7].

### *B. Datasets used*

When it comes to sentiment analysis, the contents of the dataset play a pivotal role in how the model performs in classifying the data. For example, If an African-American uses the term, it may not be considered offensive in any manner whereas if a person belonging to another race uses it, it may be considered a racial offense. The presence of such language online depending on the context of the text used presents an obstacle which must be overcome by the detection system built. To research this problem, a database containing hate speech vocabulary from various sites namely-twitter and Hatebase.org was created through manual coding [1].

Majority of the papers implementing sentimental analysis or text-based cyberbullying detection made use of textual data obtained from social media platforms like Twitter, Instagram, FormSpring.me, etc. A multiclass labelled dataset for cyberbullying detection was created by S. Salawu, J. Lumsden, and Y. He [2] to ensure that the models can detect abuse in samples that do not contain profanities. The dataset contains Twitter posts that involve a diverse vocabulary suggesting cyberbullying, online abuse, and other socially undesirable content in imbalanced distribution.

To maintain high exposure to direct and indirect forms of cyberbullying, the authors have defined and included a large range of labels for the samples obtained from Twitter-bullying, spam, profanity, sarcasm, insult, exclusion, etc. and thereby providing a detailed classification on what can be considered as bullying and offensive [2]. The classification

979-8-3503-9341-5/23/$31.00 ©2023 IEEE

of data in terms of aggression is capable of producing a higher performance, as in the case of A. Tommasel, J. M. Rodriguez and D. Godoy, who derived datasets from twitter and Facebook and revolved around topics followed by the Indian society in English text and consisted of 3 levels of aggression- Overt Aggressive (OAG), Covert Aggressive (CAG) and Non-Aggressive (NAG) [5].

### *C. Automated Detection Methods*

A variety of models have been implemented to detect cyberbullying which makes use of NLP techniques and sentiment analysis to enable machine learning models to detect aggression in texts. An artificial neural network model called Growing Hierarchical Self Organizing Map (GHSOM) developed by M. Di Capua, E. Di Nardo and A. Petrosino with clustering algorithms was used to compare unsupervised learning techniques using NLP pre-processing techniques to reduce the complexity of manual annotation of samples. They also implemented a sentiment analytic approach under the premise that the samples contained textual data of high severity. For effective prediction, they partitioned the features of a hybrid textual dataset into 4 different kinds- Syntactic features, Semantic features, Sentiment features and Social features and implemented a Growing Hierarchical self-organizing map (SOM), a popular artificial neural network which gave good performance results. The models built gave 73% accuracy when the K-means clustering algorithm on a FormSpring.me dataset and a fair accuracy of 69% on a YouTube dataset [6].

A survey conducted by studying published research and conference papers on textual cyberbullying classified the individuals involved in the act of online bullying into the victim, bystander, bully, assistant, etc. It observed that textual cyberbullying falls under the category of classification problems and that online textual harassment and bullying are elaborate in terms of context and sentiments the user expresses. For example, a positive online post may contain sarcasm which may disguise the text as "bullying" in the eyes of a detection model, or posts containing words of positivity such as "love" can be used with a sarcastic tone (e.g.- "I love how you like a clown every day.") but may be detected as "not bullying". In most cases, the samples with a high level of profanities were predicted correctly. The datasets used in the majority of the papers involved pre-processing techniques where hashtags, retweets, user tags, etc., and stop words were removed and supervised models such as Support Vector Machine (SVM), k-Nearest Neighbour (kNN), and Naive Bayes were used. Ultimately, for understanding the pattern of cyberbullying, the frequency of texts was observed and by tracking the pattern of texts, the victim and source of bullying could be identified after performing binary classification of the data. However, a limitation of this methodology would be the false labelling of samples as bullying. For evaluating performance, accuracy, precision, recall, and F1- score were calculated.

E. Sarac and S. A. ÖZEL [7] experimented with various pre-processing techniques such as stop words, lemmatization, stemming and tokenization on a FromSpring.me dataset which upon training and testing on supervised learning models revealed that stemming reduced the accuracy of the model in detecting cyberbullying. The presence of emoticons (e.g. - :D) within the text were also included as special tokens due to their capability to express emotions and feature extraction was used to retrieve relevant text along with feature selection processes, information gain and chi-square method, and classification (J48, Naïve Bayes, Ibk, and SVM). Evaluation metric, F-measure showcased that Ibk and J48 classifiers gave the best results.

Z. Hong J. Wenzhen, and Y. Guocai discusses the variation in performance with the utilization of different models like Convolutional Neural Networks (CNN), Naïve Bayes, kNN, logistic regression (LR) and many other models in textual classification. In order to counter this obstacle, the authors proposed a LAC_DNN model which takes an ensemble approach in text classification with TF-IDF weighted word2vec, TF-IDF vector space, Latent semantic index (LSI) and average word2vec for representing text-based features. The base models used for prediction using these feature representation methods are logistic regression, kNN and SVM from where the results are inputted into a neural network model, S_DNN, a small deep neural network. Evaluating the LAC_DNN model gave a high level of accuracy in classifying the records when compared with the performance of individual traditional classification models [9].

The study conducted by A. Tommasel, J. M. Rodriguez and D. Godoy [5] combines SVMs and Recurrent Neural Networks (RNN) to analyse cyber-aggression using word embeddings, irony and sentiment features, word and wide-range-of-characters. The authors analyzed the use of various features such as GloVe vectorization, TF-IDF model and SentiWordNet corpus for performing sentiment analysis on a neural network and n-gram features with TF-IDF model for SVM. The results of the study indicated that use of TF-IDF and SentiWordNet gave best performance.

In another research, Kanishk Verma , Tijana Milosevic, Keith Cortis, and Brian Davis in their research tried to compare the performance of different supervised machine learning models- SVM, Bidirectional Long Short Term Memory (Bi-LSTM) and BERT, and hate-BERT, a reoriented BERT model on a merged dataset comprising of textual data from assorted Online Social Networking (OSN) platforms like Vine, Twitter, Instagram, etc. to counter the highly imbalanced nature of the dataset they applied the concept of random over-sampling to equalize the distributions of each class. For pre-processing the dataset, different methodologies were adopted such as Glo-Ve-based vectorization for Bi-LSTM and TF-IDF vectorization for SVM. The predictions showed that the use of the merged dataset showcased a significant level of improvement in the performance of the models and exhibited that the hate-BERT model outperformed the other models [3].

Pre-processing was performed on the dataset and n-grams were formed which were further encoded via TF-IDF vectorization while keeping count of the punctuators present in each tweet. To classify the dataset, they applied decision trees, ridge logistic regression model, random forests, naive bayes and linear SVMs with 5-cross fold validation to prevent the model from overfitting. Furthermore, a one-vs-rest classification method was performed to label each tweet as hate speech, offensive or neither. The results of the models indicated that about 40% of the hate speech were wrongly classified and the model showed bias towards offensive and non-offensive tweets in comparison to the manually classified tweets. It tended to give a higher probability of correct prediction wherever derogatory terms were present [1].

To determine which methodology gives an efficient way of text classification, Y. Zhang and Z. Rao, [8] compared the performance of one-vs-rest and One-vs-one techniques on an IMDB dataset with 2 classes and a Stanford Sentiment Treebank dataset for movie reviews, consisting of 5 classes. A n-gram Bi-LSTM and Long Short Term Memory (LSTM) were run to classify samples with the 2 techniques to conclude that the use of 2-grams gave the best results when predicted with Bi-LSTM while both models obtained a good result when binary sentimental analysis was performed.

## III. Dataset Description

The dataset of size 6.88MB was taken from Kaggle [14]. It consists of 47692 records which comprise text obtained from Twitter and a column "cyberbullying type" as attributes. The attribute "cyberbullying type" is a dependent attribute that consists of 6 labels- age, other cyberbullying, religion, not cyberbullying, gender, and ethnicity. The 6 classes specify whether there is any form of cyberbullying present within each tweet in the dataset. The distribution of the records based on classes is imbalanced and several Twitter-related jargons such as retweets, user tags, links and hashtags (e.g.- #mkr) are present within the records which may be irrelevant for cyberbullying classification. Moreover, there are cases of intentional typos such as "uhhhhhh" and "BABYY", emojis and emoticons present within the tweets and tweets containing different languages apart from English, either in Romanized form or written using the respective language's keyboard (e.g.- "Dou um empurrÃ£o mu ma pessoa, isso se chama : - â€º empurrÃ£o. â€º â€º SOCIEDADE: bullying! http://tumblr.com/xun3xycfun"). For simplifying the dataset, the models built will build a vocabulary using tweets present within the train-test split and reduce the vocabulary present to simplify the dataset during preprocessing of the model.

## IV. Experimental Setup

### A. Initial Setup

The assembly and implementation of the models have been done on Google Colab. The dataset utilized is uploaded into the Colab directory by linking to the author's Google drive. For developing complete websites, inclusive of back-end and front-end coding, an online platform called Anvil was used. It provides components such as cards, text boxes, etc. which can be added onto the website and uses Python language to create functions that link the actions performed on them to the back-end program. In this project, a web template has been created and uplinked to a Google Colab file which stores the classification models.

### B. Data Preprocessing

*1) Cleaning Text:* Different models require different types of preprocessing based on the type of data the model requires as input, which is numerical data, and what kind of data structure it must be constructed in. For general preprocessing of the dataset, a function "preprocessor" was built which calls upon other related functions to clean the dataset through NLP techniques. The replace() function in Python has been utilized for the removal of HTML tags (&lt, &amp, etc.), punctuation, hashtags, user mentions, reposts, URLs, and special characters ($, %. Etc.) by inserting blank spaces in their stead. The number of blank spaces and any numbers present are also removed and replaced with a single black character to reduce the length of the text. Since many of the samples contain abbreviated forms of certain words like the use of "can't" for cannot and the suffix 'in' to shorten the verbs with the suffix 'ing', a pip package, contractions, has been imported to expand the abbreviated terms to assist the removal of stop words, words that are frequently used in a language. The stop words were imported from the NLTK package and removed using a linear search on each record.

*2) Lemmatization:* A customized lemmatization function that gives special consideration to verbs, adjectives, and adverbs were built to homogenize the vocabulary present in the records. Due to a large number of records present, the duration of preprocessing takes a while for lemmatizing, which has been sped up by using Swifter, a python package applicable to pandas data frame which helps decrease the processing time.

*3) Uniformity of records:* To maintain consistency in the length of the tweets, a statistical bar plot was drawn and tweets having at most length 3 and more than 93 were discarded. An imbalance present in the dataset was resolved through random oversampling technique.

*4) Train-test split and target variable:* The train and test samples required for each model has been obtained via a ratio of 75:25 respectively for performing a hold-out validation. Furthermore, in order to transform the textual data into numeric form for processing, CountVectorization and TF-IDF transformation has been completed. The n-gram range of 2-4 for all traditional models except the LSTM models has been varied with respect to their best individual performance for evaluate their results. The class labels are considered the independent attribute while the tweets are the dependent attribute to help the models distinguish the text on a clarified basis.

## V. Methodology

The models listed below has been constructed and linked to the web app template built on Anvil, it utilizes the established connection with the Colab platform to retrieve the input text from the website and provide the predicted labels as output on the website.

### A. Multinomial Naive Bayesian Model

Naive Bayes classifier [16] is an algorithm that implements the concept of naive Bayes theorem by calculating the likelihood each word has per label and assigns a specific label based on the highest resultant probability. Due to its ability to predict discrete labelled data, it falls under the category of supervised machine learning and is applied in fields. In this study, a multinomial naive Bayes model is built, which is applicable for sentiment analysis in texts.

### B. Linear SVM

Support vector Machine (SVM) [16] is a linear classification model which operates by plotting data points on a graph and labelling the data points based on a decision boundary. A one-vs-one classifier, a type of classifier that builds k(k-1)/2 number of linear SVM models and predicts class labels by pitting one label against every other label individually. The data has been fitted using a pipeline which performs the required processing all at once.

### C. *LSTM and Bi-LSTM*

LSTM [16] is a recurrent neural network which is capable of storing useful data for a short period of time to identify sequences present in the presented data. A RNN was built using Pytorch which is capable of implementing LSTM and bidirectional LSTM by changing a parameter named 'bi_lstm'. The model consists of one embedding layer, a LSTM layer, a fully connected layer and a SoftMax layer. The hidden layers, which present the outputs for the upcoming layers and cell states of the LSTM are initialized through a function present within the class created for the LSTM model. Since complex models take a longer period of time to process in comparison to simpler models, the model is run in a GPU enabled environment. An embedding matrix was created using a customized Tokenize() function and GloVe word embeddings, to capture the relations between unique words in the training dataset. The embedding matrix is used as weights for the embedding layer. The data is converted into TensorFlow datasets and loaded for running training and validation loops using unigram format.

### D. *Logistic Regression*

Logistic regression [16] is one of the frequent supervised methods applied in classification. It performs classification on the basis of probabilistic values, by estimating the maximum likelihood of a label being correct and normalizes the value into 0 or 1. The LR model defined uses multinomial regression to classify the records into appropriate labels.

### E. *Ensemble Learning*

Ensemble learning aims to combine the results of models to get the best possible accuracy and improve the predictiveness of the constructed models. There are different types of ensemble learning, out of which the following have been implemented in this project:

*1) XGBoost Classifier:* The XGBoost [16] classifier uses the hist-tree method for performing boosting technique.

*2) Stack Ensemble Learning:* Stack ensemble learning combines the predictions of several weak classifiers as input to a final classifier to combine the results from the previous classifiers and generate suitable labels for the tested data. The stacking model built for cyberbullying classification here involves the use of Naive Bayesian model and SVM as estimators, the first layer of models and LR as the final layer.

## VI. Performance Analysis

### A. *Evaluation metrics*

To evaluate how well the models execute and how long each of them takes to complete, a confusion matrix and classification report are drawn using the original labels of the test set and the predicted classes. The accuracy, weighted precision, weighted recall, weighted F1-score and the amount of time the model took from pre-processing to prediction are given in Table I. In terms of accuracy, the LSTM and Bi-LSTM model showcased the best performance by 92% which means that the model may have overfit a little bit. The XGBoost classifier gave the lowest result 75%. The accuracies of SVM, multinomial Bayes classifier and logistic regression lie within the range 80% to 90% indicating that the models perform well. The performance metric weighted precision has been adapted to analyze whether the models have predicted the classes correctly according to their cyberbullying labels. The highest precision of true positives, data which was predicted correctly, was given by LSTM and Bi-LSTM followed by XGBoost and Naive Bayes model. Taking individual precision values of each class provided by the classification report for each model, it can be observed that the quality of prediction for the labels 'age',' ethnicity' and 'gender' were fairly high for all models while 'religion' and 'not cyberbullying' were comparatively lower than that of the others.

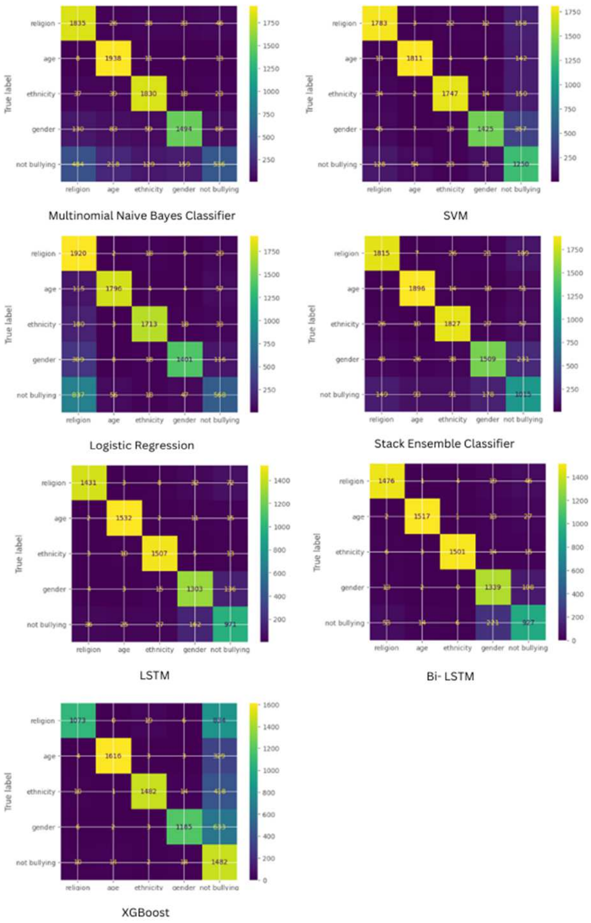

Fig. 1. Confusion matrix of each model

TABLE I. Evaluation metrics and runtime of each model.

| Evaluation metrics | Models | | | | | | |
|---|---|---|---|---|---|---|---|
| | ***Multinomial Naïve Bayes*** | ***SVM*** | ***LR*** | ***LSTM*** | ***Bi-LSTM*** | ***Stack Ensemble*** | ***XGBoost*** |
| Accuracy | 0.82 | 0.86 | 0.80 | 0.92 | 0.92 | 0.87 | 0.75 |
| Weighted Precision | 0.82 | 0.88 | 0.83 | 0.92 | 0.92 | 0.87 | 0.88 |
| Weighted Recall | 0.82 | 0.86 | 0.78 | 0.92 | 0.92 | 0.87 | 0.75 |
| Weighted F1-score | 0.79 | 0.87 | 0.78 | 0.92 | 0.92 | 0.87 | 0.77 |
| Runtime | 0.15 | 13 | 1.8 | 8.94 | 10.43 | 60 | 24.2 |

The ensemble model and SVM models gave the lowest scores for ‘not cyberbullying’ while the Bi-LSTM model gave the highest (0.83) among the models. For the label ‘religion’, the highest precision was obtained by LSTM (0.97) while LR displayed poor performance in terms of precision.

For weighted recall, an overall score that indicates the number of labels predicted according to their original labels, the Bi-LSTM and LSTM models performed best followed by the ensemble model. The recall for each label predicted via Bi-LSTM gave slightly higher scores than that of LSTM. However, the LSTM model predicted a larger number of records correctly for the label ‘not cyberbullying’ when compared to that of LSTM.

Since the accuracy of the model does not consider the records which were falsely classified, the weighted F1-score of each model has been taken into account. As expected, the LSTM and Bi-LSTM models gave the highest F1 scores proving that the models excel at classifying cyberbullying. The SVM model and stack ensemble model also gave good scores indicating that they perform well without any signs of overfitting. The runtime taken by each model was also recorded in minutes to evaluate how long each model would take. The shortest period of execution was for the Naive Bayes classifier (0.15 minutes) which is a huge contrast to that of the ensemble model which took an hour. The rest of the models ran for various periods as listed in table I. The Bi-LSTM and LSTM models performed for a reasonable period of 10.43 and 8.94 minutes respectively as the tokenization and GloVe vectorization of data took some time.

### *B. Testing on Unseen Data*

To observe the performance of the classifiers on unseen data, a couple of random social media posts were drawn from the internet and tested on the website. Table II shows the length of the tested text and the labels predicted for that particular text by each of the models. With increase in the length of the text, the models give more defined results. It can be observed from the fifth post in the table, that almost all the models gave the “religion” label corresponding to religious hate, except the Naive Bayes model which labelled the text as “ethnicity”-related cyberbullying, both of which seem appropriate for the text. For shorter posts, the Stack Ensemble model gave correct results in comparison to the others, as evident in the first record in the table. The dearth of direct racism-related term, the fourth record was classified as non-cyberbullying by most classifiers and “religion” by Naive Bayes classifier while the LSTM and Bi-LSTM models predicted “ethnicity” correctly. Although racism is depicted clearly in the text from a human perspective, the models found the text to be ambiguous. The Bi-LSTM and LSTM models were able to detect all forms of cyberbullying correctly apart from being confused between the labels “Age” and “Ethnicity” for record 1 in table II.

TABLE II. TESTING ALL THE MODELS ON UNSEEN DATA

| **Sample data** | **Model output** | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ***Text Length (words)*** | ***Multinomial Naïve Bayes*** | ***SVM*** | ***LR*** | ***Stack Ensemble*** | ***LSTM*** | ***Bi-LSTM*** | ***XGBoost*** |
| I couldn’t care less about Indians were it not for the fact that they are actively sabotaging China. Mask off then. Indians are genuinely bad people. The vast majority of them. | 31 | Not bullying | Not bullying | Not bullying | Ethnicity | Age | Ethnicity | Not bullying |
| RT @kohfuckyourself I’m not sexist, but Feminists make me sick in how they go about fighting for equality. Sorry. | 19 | Gender | Gender | Gender | Gender | Ethnicity | Ethnicity | Gender |
| With make-up on, she totally looks like a Russian elf | 25 | Age | Not bullying | Not bullying | Age | Age | Age | Not bullying |
| If you play with her, your skin will turn black. Tsk tsk. | 12 | Religion | Not bullying | Not bullying | Not bullying | Ethnicity | Ethnicity | Not bullying |
| ::You Zionist Jewbastard Khazar Turks just love filibusters that draw out this tragedy to no conclusion. That’s right, only YOU are allowed a say on the issue. YOU have the right to editorialise anything to YOUR content, media mogul jackasses! Stay out of London, New York, Washington and Hollywood! Helen Clark … I’ll take nukes signed by each and every Jew of the Manhattan Project and level you to nothing; in a eulogy …….You are savages without civilisation and bloodsucking leeches holding onto hosts as all viruses do; so you are fake friends! No more Bugsy Siegel Hollowcau$e Industry and kosher racketeering! ::GENOCIDIST DAVID SLEW GOLIATH OF PALESTINE! REJECTED ONES, YOU HAVEN’T MONOPOLY ON SUFFERING! | 414 | Ethnicity | Religion | Religion | Religion | Religion | Religion | Religion |

## VII. CONCLUSION

Research was conducted to analyze the performance of different models in predicting cyberbullying on social media. For this purpose, several research and conference papers related to text classification were reviewed and an appropriate dataset was chosen from Kaggle. Pre-processing was performed on the dataset using NLP techniques and the text data was converted into a numeric format appropriate for each of the models built, namely- SVM, Multinomial Bayesian classifier, LR, XGBoost, LSTM, BI-LSTM and Stacking Ensemble classifier. The models were evaluated using the prepared data and compared with one another using evaluation metrics, out of which the LSTM models showed the best performance. There was also a significant decline in the amount of time spent fitting and prediction of data by the both the LSTM models with similar outputs. An additional observation from the performance of models on unseen text is that there are instances of multiple classes for ambiguous texts which needs to be addressed. A web application was built using the models and a web app creator named Anvil to input a text and display the different labels assigned to it by each of the classifiers to display the variation in output given by each model for a piece of unseen text.

## VIII. Future Scope

With respect to the results of the comparison conducted, the future research shall include the prediction of multiple labels. It shall make use of Large Language Models and hybrid learning to evaluate cyberbullying classification using different metrics. With respect to runtime, the time taken to fit the model can also be reduced by fine-tuning the model and adjusting the data size. It is also vital that the models are taught to identify the situation through the use of emojis, abbreviated text forms such as "KYS", which means "Kill Yourself" and text variations (typing "die" as "di3"). This can be achieved by inclusion of a variety of samples within the dataset. As the sensitive nature of people increases over social media, it is important to find an evolving solution to the destructive act of cyberbullying.

## References

[1] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated Hate Speech Detection and the Problem of Offensive Language", ICWSM, vol. 11, no. 1, pp. 512-515, May 2017.

[2] S. Salawu, J. Lumsden, and Y. He, "A Large-Scale English Multi-Label Twitter Dataset for Cyberbullying and Online Abuse Detection," in Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021), Association for Computational Linguistics, Jul. 2021, pp. 146–156. Accessed: Mar. 22, 2023. [Online]. Available: https://aclanthology.org/2021.woah-1.16

[3] K. Verma, T. Milosevic, K. Cortis, and B. Davis, "Benchmarking Language Models for Cyberbullying Identification and Classification from Social-media Texts," in Proceedings of the First Workshop on Language Technology and Resources for a Fair, Inclusive, and Safe Society within the 13th Language Resources and Evaluation Conference, European Language Resources Association, Jun. 2022, pp. 26–31. Accessed: Mar. 26, 2023. [Online]. Available: https://aclanthology.org/2022.lateraisse-1.4

[4] S. Salawu, Y. He and J. Lumsden, "Approaches to Automated Detection of Cyberbullying: A Survey," in IEEE Transactions on Affective Computing, vol. 11, no. 1, pp. 3-24, 1 Jan.-March 2020, doi: 10.1109/TAFFC.2017.2761757.

[5] A. Tommasel, J. Rodriguez, and D. Godoy, "Textual Aggression Detection through Deep Learning," in Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018), Association for Computational Linguistics, pp. 177–187.

[6] M. Di Capua, E. Di Nardo and A. Petrosino, "Unsupervised cyber bullying detection in social networks," 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 2016, pp. 432-437, doi: 10.1109/ICPR.2016.7899672.

[7] S. A. Özel and E. Saraç , "Effects of Feature Extraction and Classification Methods on Cyberbully Detection", Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi, vol. 21, no. 1, pp. 190-200, Apr. 2017, doi:10.19113/sdufbed.20964

[8] Y. Zhang and Z. Rao, "n-BiLSTM: BiLSTM with n-gram Features for Text Classification," 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC). IEEE, Jun. 2020. Doi: 10.1109/itoec49072.2020.9141692.

[9] Z. Hong, J. Wenzhen, and Y. Guocai, "An Effective Text Classification Model Based on Ensemble Strategy," in Journal of Physics: Conference Series, IOP Publishing Ltd, 2019, p. 012058.

[10] S. Atske, "Teens and Cyberbullying 2022," Pew Research Center: Internet, Science & Tech, Dec. 15, 2022. https://www.pewresearch.org/internet/2022/12/15/teens-and-cyberbullying-2022 (accessed Mar. 22, 2023).

[11] "SOSNet: A Graph Convolutional Network Approach to Fine-Grained Cyberbullying Detection," Proceedings of the 2020 IEEE International Conference on Big Data (IEEE BigData 2020), December 10-13, 2020.

[12] A. Tam, "LSTM for Time Series Prediction in PyTorch," Machine Learning Mastery, Aug. 08, 2023. https://machinelearningmastery.com/lstm-for-time-series-prediction-in-pytorch/ (accessed Mar. 26, 2023).

[13] "sklearn.multiclass.OneVsOneClassifier," scikit-learn. https://scikit-learn.org/stable/modules/generated/sklearn.multiclass.OneVsOneClassifier.html (accessed Mar. 12, 2023).

[14] "Cyberbullying Classification," www.kaggle.com. https://www.kaggle.com/datasets/andrewmvd/cyberbullying-classification

[15] S. Yıldırım, "15 Must-Know Machine Learning Algorithms," Medium, Jan. 06, 2021. https://towardsdatascience.com/15-must-know-machine-learning-algorithms-44faf6bc758e

[16] R. K. Mishra, H. Raj, S. Urolagin, J. A. A. Jothi, and N. Nawaz, "Cluster-Based Knowledge Graph and Entity-Relation Representation on Tourism Economical Sentiments," Applied Sciences, vol. 12, no. 16, p. 8105, Aug. 2022, doi: 10.3390/app12168105.