International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2454-9150 Vol-10, Issue-01, Apr 2024

Cyberbullying Detection using SVM Algorithm

Mr. Prathamesh Ahire[1], Mr. Pushpak Deore[2], Ms. Pranita Panpatil[3], Mr. Chinmay Shinde[4], Dr. P. D. Halle[5]

[1,2,3,4]UG Student, [5]Asst. Dr, dept. of Information Technology SKN Sinhgad Institute of Technology & Science, Lonavala, Maharashtra, India. [1]prathameshahire0123@gmail.com, [2]deorepushpakk@gmail.com, [3]pranita2020p@gmail.com, [4]shindechinmay1234@gmail.com, [6]hallepriyanka2011@gmail.com

Abstract - Cyberbullying has emerged as a significant societal issue prevalent on the internet, impacting both adolescents and adults alike. Its detrimental effects include instances of suicide and depression among victims. Consequently, there is an escalating necessity for regulating content across social media platforms. This study endeavors to address cyberbullying through the utilization of Natural Language Processing (NLP) and Machine Learning (ML) techniques, employing data sourced from two distinct forms of cyberbullying: hate speech tweets from Twitter and personal attack comments from Wikipedia forums. The research aims to construct a model for detecting cyberbullying in textual data. Specifically, three distinct methods for feature extraction and four classifiers are scrutinized to ascertain the optimal approach. The evaluation of the model indicates that for tweet data, accuracies surpassing 90% are achieved, while for Wikipedia data, accuracies exceed 80%. This research contributes to the ongoing efforts in combating cyberbullying through advanced computational techniques.

Keywords: Cyberbullying, Hate speech, Personal attacks, Machine learning, Feature extraction, Twitter, Wikipedia

I. INTRODUCTION

In contemporary society, technology has assumed a paramount role in our daily lives, with the internet undergoing significant evolution. The rise of social media platforms has become particularly pronounced. However, like many other advancements, misuse and abuse inevitably emerge. Cyberbullying has become increasingly prevalent in this digital landscape.

Social networking sites serve as invaluable tools for interpersonal communication. Despite their widespread adoption, there is a disturbing trend of individuals engaging in unethical and immoral behavior online. This behavior often manifests as bullying, occurring predominantly among teenagers and young adults. The online environment complicates the interpretation of intent behind individuals' actions; what may be construed as harmless jest could conceal more malicious intentions. Cyberbullying encompasses a spectrum of behaviors, including harassment, threats, embarrassment, and targeting of individuals.

Tragically, the repercussions of cyberbullying extend beyond the virtual realm, with real-life consequences such as threats and, in extreme cases, suicide. Preventative measures are imperative to curb such harmful activities. Strategies may include swift interventions, such as suspending or terminating the accounts of individuals found engaging in offensive behavior online.

In essence, cyberbullying encompasses various forms of harassment, ranging from casual jest to premeditated attacks, perpetrated through digital mean.

II. LITERATURE SURVEY

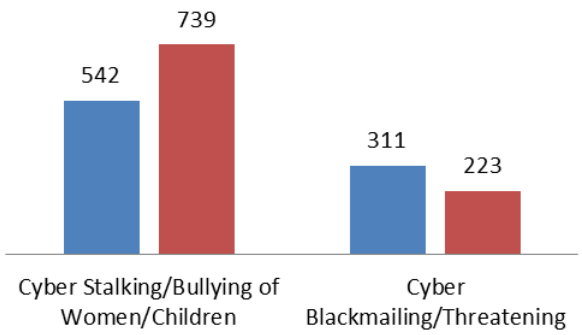

Fig. 1. Cyberbullying cases in India 2021-2022

Considerable research has been conducted to address the challenge of detecting cyberbullying on social networking sites. Ting, I-Hsien [1] employed an approach integrating keyword matching, opinion mining, and social network analysis, achieving a precision of 0.79 and a recall of 0.71 from datasets sourced from four websites. Patxi Galán-García et al. [2] proposed a hypothesis suggesting that individuals engaging in cyberbullying, known as trolls, often maintain a real profile to monitor reactions to their fake profiles on social networking sites. Their machine learning-

 DOI : 10.35291/2454-9150.2024.0170

based approach aimed to identify such profiles by analyzing features and tweets from closely related profiles. The method, applied to 1900 tweets from 19 profiles, achieved a 68% accuracy in author identification. Subsequently, this method was successfully utilized in a case study at a school in Spain to identify the real owners of suspected cyberbullying profiles. However, this approach still faces limitations, such as cases where trolling accounts lack real profiles to deceive detection systems or when experts alter writing styles and behaviors to evade pattern recognition. Addressing the challenge of changing writing styles requires more sophisticated algorithms.

Mangaonkar et al. [3] proposed a collaborative detection method involving interconnected detection nodes, each employing either different or identical algorithms and datasets. Results from these nodes were combined to enhance detection accuracy. P. Zhou et al. [4] introduced a B-LSTM technique based on concentration, while Banerjee et al. [5] utilized KNN with new embeddings, achieving a precision of 93%. Kelly Reynolds, April Kontostathis, and Lynne Edwards [6] introduced a dataset from Formspring, a forum for anonymous questions and answers, which yielded a recall of 78.5% using machine learning algorithms and oversampling due to imbalances in cyberbullying posts.

Jaideep Yadav, Kumar, and Chauhan [7] employed the BERST language model developed by Google, which generates contextual embeddings for classification. Their model achieved an F1 score of 0.94 on Formspring data and 0.81 on Wikipedia data. Maral Dadvar and Kai Eckert [8] trained deep neural networks on Twitter, Wikipedia, and Formspring datasets, subsequently applying the model to a YouTube dataset, achieving an F1 score of 0.97 using a Bidirectional Long Short-Term Memory (BLSTM) model. Sweta Agrawal and Amit Awekar [9] also utilized similar datasets to train deep neural networks, with a particular focus on swear words as features. They investigated the variation in vocabulary across various social media platforms. Yasin N. Silva, Christopher Rich, and Deborah Hall [10] developed BullyBlocker, a mobile application designed to inform parents of cyberbullying activities against their child on Facebook. This application assessed warning signs and vulnerability factors to calculate a probability measure of being bullied.

III. METHODOLOGY

Cyberbullying detection is solved in this project as a binary classification problem where we are detecting two majors form of Cyberbullying: hate speech on Twitter and Personal attacks on Wikipedia and classifying them as containing Cyberbullying or not.

Fig. 2 describes the methodology used for solving the problem which is applied on both the datasets.

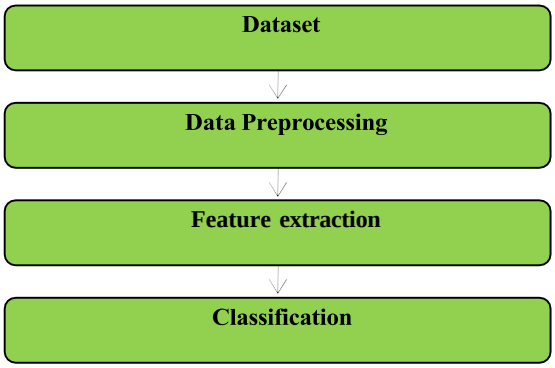

Fig. 2. Methodology

a) DATASET

A. Twitter Dataset

The Twitter dataset utilized in this study is amalgamated from two distinct sources focusing on hate speech:

1. The Hate Speech Twitter Dataset by Waseem, Zeerak, and Hovy [11], comprising 17,000 tweets annotated for instances of sexism or racism. It is noted that 5,900 tweets were lost due to account deactivation or tweet deletion.

2. The Hate Speech Language Dataset by Davidson, Thomas, Warmsley, Dana, Macy, Michael, and Weber, Ingmar [12], which consists of 25,000 tweets gathered through crowdsourcing.

Combining these two datasets results in a total of 35,787 tweets for analysis, as illustrated in Figure 3. For the purposes of this study, 70% (25,050) of the dataset is allocated for training data, while the remaining 30% (10,737) is designated for testing data.

B. Wikipedia Dataset

The Wikipedia dataset, as curated by Wulczyn, Thain, and Dixon [13], comprises 1 million comments annotated for instances of personal attacks. For the purposes of this analysis, a subset of 40,000 comments is selected from the dataset, of which 13,000 comments are identified as instances of cyberbullying involving personal attacks. These comments are extracted from conversations among Wikipedia page editors and were labeled by ten annotators via CrowdFlower.

Similar to the Twitter dataset, the Wikipedia dataset is partitioned using the same split ratio, allocating 70% (28,000) of the comments for training data and 30% (12,000) for testing data. The distribution of this dataset is depicted in Figure 4.

 DOI : 10.35291/2454-9150.2024.0170

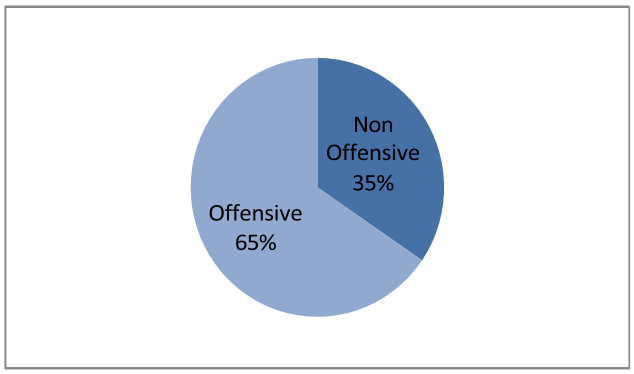

Fig. 3. Distribution of Tweets in Twitter Dataset

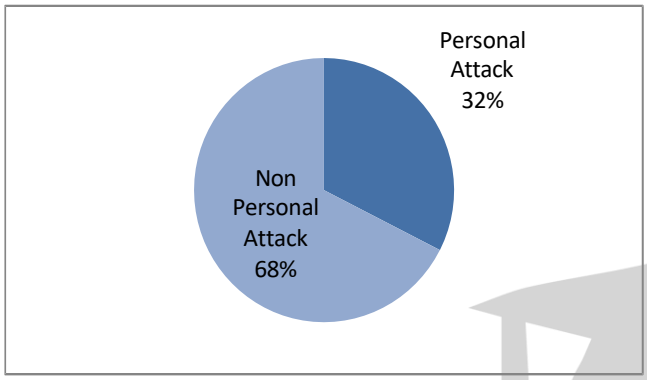

Fig. 4. Distribution of Comments for Wikipedia Dataset

b) DATA PREPROCESSING

A data processing pipeline, as depicted in Figure 5, is employed for both datasets. Initially, all text data is converted to lowercase to ensure uniformity. Subsequently, certain contractions, such as "what's" or "can't," are expanded to their full forms, such as "what is" or "cannot." Additionally, all punctuation marks are removed using the string library. Following these initial steps, various Natural Language Processing (NLP) techniques are applied using the Natural Language Toolkit (NLTK):

- Tokenization: This process involves breaking raw text into meaningful tokens, such as words or sentences. In this project, Regex Tokenizer is utilized, which segments tokens based on a specified regular expression pattern. For instance, the regular expression '\w+' extracts alphanumeric tokens.

- Stemming: Stemming is the process of reducing words to their root forms or stems. NLTK offers various stemmers, including Porter Stemmer, Lancaster Stemmer, Snowball Stemmer, and Regexp Stemmer. The Porter Stemmer is employed in this project to standardize word forms and recognize similarities between related words.

- Stop Word Removal: Stop words, such as "what," "is," "at," and "a," are common words that add little semantic value to sentences. NLTK provides a list of English stop words for filtering out irrelevant words from text data. Removing stop words is particularly beneficial for training machine learning and deep learning models, as it improves performance by eliminating noise.

c) FEATURE EXTRACTION

Feature extraction plays a pivotal role in Natural Language Processing, as text data cannot be directly classified by classifiers and must be converted into numerical data. Each document, whether a tweet or a comment, is represented as a vector, with these vectors serving as input for classification. This project investigates three feature extraction methods: Bag of Words, TF-IDF, and Word2Vec.

A. Bag of Words Model

The Bag of Words (BoW) model is a straightforward approach to feature extraction from documents, focusing on the occurrence of words within each document. This model comprises two essential components:

- Vocabulary: A collection of words (tokens) derived from all documents.

- Feature Measurement: A method for quantifying the presence of these words as features in each document.

The BoW model disregards the order of words in a document, hence the term "bag," emphasizing the focus on words themselves rather than their sequence. The rationale behind this method is that documents with similar content tend to share common words.

The BoW model follows these steps:

1. Vocabulary Design: A vocabulary is constructed from all documents, comprising all words (tokens) or a subset of top frequency tokens. Additionally, features can be extracted using different combinations of words per feature, such as unigram, bigram, or n-gram models.

2. Document Transformation: Once the vocabulary is established, documents are transformed based on the vocabulary using a feature measurement method. Two common approaches include binary representation, where features are represented as either 1 or 0 depending on their presence in a document, and frequency representation, where features are represented by their occurrence frequencies.

While the Bag of Words model is effective for sentiment analysis, it has limitations, such as disregarding word context and order, which may impact performance in certain cases. Additionally, designing a vocabulary becomes challenging with large datasets due to the proliferation of features. For instance, the sentences "Is it interesting" and "It is interesting" convey different meanings, emphasizing the importance of context.

B. F-IDF Model

Tf-Idf method is similar to the bag of words model since it uses the same way to create a vocabulary to get its features. TF-IDF addresses a problem not seen much in the corpus, but is important for better extraction of features. The value

 DOI : 10.35291/2454-9150.2024.0170

of Tf-Idf increases with the increase in frequency of a word in same document and decreases with decrease in frequency of documents that have the word in the corpus.

Both of these methods use forward and back propagation to train the neural networks and find the best parameters. For each document then a feature vector can be created by concatenating and combining all word vectors in that document. Combination of word vectors can be done by summation or by averaging all word vectors. Selection between the both is based on data.

d) CLASSIFICATION

After obtaining feature vectors for the training data through the aforementioned feature extraction methods, the testing data is transformed using the same scheme without undergoing fitting on the vectorizers or training on the Word2Vec model. Subsequently, the following classifiers are trained and tested using the training data:

A. Support Vector Machine (SVM)

The Support Vector Machine (SVM) theorem is employed to establish a hyperplane that delineates boundaries between data points in a high-dimensional feature space. To optimize the margin value, the hinge function serves as one of the most effective loss functions. In this scenario, Linear SVM is utilized, which is particularly suitable for linearly separable data. In the event of zero misclassification, indicating accurate prediction of the class of data points by our model, adjustments solely involve modifying the gradient from the regularization arguments. However, in cases of misclassification, where our model erroneously predicts the class of a data point, adjustments are made by incorporating the reduction with the gradient update regularization.

B. Random Forest

A random forest is composed of numerous individual decision trees, each of which predicts a class for given query points. The final result is determined by the class with the maximum votes among all the trees. Decision trees serve as the building blocks for random forests, providing predictions based on decision rules learned from feature vectors. By aggregating the predictions of these uncorrelated trees, random forests offer a more accurate decision for classification or regression tasks.

C. Multi-Layered Perceptron

Multi-Layered Perceptrons (MLPs) are a type of Artificial Neural Networks (ANNs) that consist of at least three layers: one input layer, one output layer, and one or more hidden layers. Each node in these layers computes an activation value using an activation function during forward propagation. Backpropagation is then employed to train the weights used in the neural network. MLPs are typically utilized when the data is not linearly separable. Commonly used activation functions include ReLU (Rectified Linear Unit) and sigmoid. The sigmoid function, similar to the hyperbolic tan function (tanh), produces values between -1 and 1. ReLU is defined as f(x) = max(0, x). MLPs can be constructed and trained using frameworks such as Keras.

IV. EXPERIMENTS AND RESULTS

Google colab was used for the experiments. For each classifier the following parameters were evaluated on the test sets

- Accuracy(A): is defined as no of correct predictions divided by total number of predictions.

$$A = \frac{\text{True positives}}{\text{Size of dataset}}$$

- Precision(P): Out of all the positive predictions by the classifier how many are actually positive.

$$P = \frac{\text{True positives}}{\text{True + False Positives}}$$

- Recall(R): Out of all the positive inputs how many were predicted positive.

$$R = \frac{\text{True positives}}{\text{True Positives + False Negative}}$$

The following configurations are used for the feature selection methods for both datasets used:

- Bag of Words Model: Top 10000 features out of Unigrams, Bigram and Trigram features were selected based on frequency.
- TF-IDF Model: Same as Bag of words model
- For the word2vec model both the skips-gram and Common Bag of Words(CBOW) model were trained. 200 features from both models were combined to get 400 features for each word embedding and for each document summation was used to generate document vector. word2vec was trained on the training sets for 30 epochs with a window of 5 words.

The classifiers were loaded through sklearn library except the Multi Layer Perceptrons which were made in Keras. Two MLPs were used: one for Bag of Words and tfidf for 10000 feature input and other for 400 feature input of Word2vec. The Classifiers used are Linear SVM (SVC), Random Forest Classifier (RF), Logistic Regression (LR) and Multi Layered Perceptron (MLP).

Tables 1 and 2 show results for Twitter and Wikipedia dataset respectively.

The Twitter dataset which contained tweets related to Hate speech show F- measures above 0.9 for all three feature selection methods. The values for Word2Vec model are a

 DOI : 10.35291/2454-9150.2024.0170

bit less but are ideal considering it used 400 features instead of other methods using 10000. TF-IDF method combined with Linear SVM gives best recall and F-measure.

For the Wikipedia dataset which contained comments with Personal attacks it shows F-measures only around 0.8 for all models. The TF-IDF with Linear SVM still get the best F- measure but Word2Vec with Multi Layered Perceptron gives better recall.

V. CONCLUSION

Cyber bullying across internet is dangerous and leads to mishappenings like suicides, depression etc and therefore there is a need to control its spread. Therefore cyber bullying detection is vital on social media platforms. With avaibility of more data and better classified user information.

Our investigation into cyberbullying across internet platforms has highlighted its grave consequences, including incidents such as suicides and depression. This underscores the urgent need for effective measures to control its spread. Through this research, we emphasize the critical importance of cyberbullying detection, particularly on social media platforms where such activities proliferate.

In our study, we proposed an innovative architecture for cyberbullying detection aimed at combating this pervasive issue. We focused on two primary datasets: hate speech data from Twitter and personal attacks on Wikipedia. Our findings demonstrate the effectiveness of employing natural language processing techniques, particularly with basic machine learning algorithms, to achieve accuracies exceeding 90% in detecting hate speech.

While hate speech detection benefited significantly from bag-of-words (BoW) and TF-IDF models due to the prevalence of profanity, the detection of personal attacks presented a more nuanced challenge. Despite the absence of explicit sentiment markers in personal attack comments, our exploration of Word2Vec models leveraging contextual features yielded promising results. This approach, particularly when combined with multi-layered perceptrons, showcased comparable performance across datasets with fewer features.

In conclusion, our research underscores the importance of leveraging advanced computational techniques for cyberbullying detection. By addressing the unique characteristics of different forms of cyber aggression, we can develop more robust and effective strategies for safeguarding online communities. Moving forward, continued advancements in machine learning and natural language processing offer promising avenues for enhancing cyberbullying mitigation efforts and promoting safer digital environments.

VI. FUTURE SCOPE

While this research provides valuable insights into the detection of cyberbullying through natural language processing and machine learning techniques, there are several avenues for future exploration and enhancement:

1. **Enhanced Feature Selection**: Investigate advanced feature selection methods beyond Bag of Words, TF-IDF, and Word2Vec. Techniques such as contextual embeddings, attention mechanisms, and domain-specific features could be explored to improve model performance.
2. **Deep Learning Architectures**: Explore more complex deep learning architectures, such as recurrent neural networks (RNNs), convolutional neural networks (CNNs), or transformer-based models like BERT, to capture intricate patterns and semantics in textual data more effectively.
3. **Fine-tuning Models:**Experiment with fine-tuning pre-trained language models on cyberbullying detection tasks. Transfer learning from large-scale language models like GPT and BERT could potentially enhance model generalization and performance on specific cyberbullying datasets.
4. **Multimodal Approaches**: Investigate multimodal approaches that incorporate textual, visual, and auditory cues for cyberbullying detection. Combining information from multiple modalities could lead to more robust and accurate detection systems.
5. **Real-time Detection**: Develop real-time cyberbullying detection systems capable of monitoring social media platforms and online forums continuously. Integration with social media APIs and advanced streaming processing techniques could enable timely intervention and prevention of cyberbullying incidents.
6. **Evaluation Metrics**: Explore additional evaluation metrics beyond precision, recall, and F-measure to assess model performance comprehensively. Metrics such as specificity, sensitivity, and area under the ROC curve (AUC) could provide a more nuanced understanding of model capabilities.

By pursuing these avenues for future research, we can advance the state-of-the-art in cyberbullying detection and contribute to creating safer and more inclusive online environments for individuals of all ages.

REFERENCE

[1]https://ieeexplore.ieee.org/document/9163353

[2] https://ieeexplore.ieee.org/document/10122521

[3] https://ieeexplore.ieee.org/document/10138186

[4] https://www.researchgate.net/publication/355116964_

 DOI : 10.35291/2454-9150.2024.0170

Cyberbullying_Behaviour_A_Study_of_Undergraduate_University_Students

[5] https://ieeexplore.ieee.org/document/9718597

[6]. Halle, P.D., Shiyamala, S.and Rohokale, Dr.V.M. (2020) "Secure Directionfinding Protocols and QoS for WSN for Diverse Applications-A Review," International Journal of Future Generation Communication and Networking, Vol. 13 No. 3 (2020) Available at: https://sersc.org/journals/index.php/IJFGCN/article/view/26983. (Web of Science).

[7]. Halle, P.D. and Shiyamala, S. (2020) "Trust and Cryptography Centered Privileged Routing Providing Reliability for WSN Considering Dos Attack Designed for AMI of Smart Grid," International Journal of Innovative Technology and Exploring Engineering. Blue Eyes Intelligence Engineering and Sciences Engineering and Sciences Publication-BEIESP. doi:10.35940/ijitee.b7449.019320.

[8]. Halle, P.D. and Shiyamala, S. (2021) Ami and its wireless communication security aspects with QOS: A Review, SpringerLink. Springer Singapore. Available at: https://link.springer.com/chapter/10.1007/978-981-15-5029-4_1 (Scopus: Conference Proceeding Book Chapter).

[9]. Halle, P.D. and Shiyamala, S. (2022) "Internet of things enabled secure advanced metering infrastructure protocol for smart grid power system" SN Computer Science. (Scopus under review).

 DOI : 10.35291/2454-9150.2024.0170