**Detecting A Twitter Cyberbullying Using Machine learning**

Social media is a platform where many young people are getting bullied. As social networking sites are increasing, cyberbullying is increasing day by day. To identify word similarities in the tweets made by bullies and make use of machine learning and can develop an ML model automatically detect social media bullying actions. However, many social media bullying detection techniques have been implemented, but many of them were textual based. The goal of this paper is to show the implementation of software that will detect bullied tweets, posts, etc. A machine learning model is proposed to detect and prevent bullying on Twitter. Two classifiers i.e. SVM and Naïve Bayes are used for training and testing the social media bullying content. Both Naive Bayes and SVM (Support Vector Machine) were able to detect the true positives with 71.25% and 52.70% accuracy respectively. But SVM outperforms Naive Bayes of similar work on the same dataset. Also, Twitter API is used to fetch tweets and tweets are passed to the model to detect whether the tweets are bullying or not.

**EXISTING SYSTEM:**

Cyber bullying is the use of technology as a medium to bully someone. Although it has been an issue for many years, the recognition of its impact on young people has recently increased. Social networking sites provide a fertile medium for bullies, and teens and young adults who use these sites are vulnerable to attacks. Through machine learning, we can detect language patterns used by bullies and their victims, and develop rules to automatically detect cyber bullying content.

**DISADVANTAGES OF EXISTING SYSTEM:**

* Hence the results of producing with such kind of results are not satisfying.
* Datasets are not sufficient to predict better results.

**Algorithms:** KNN.

**PROPOSED SYSTEM:**

In this paper, a solution is proposed to detect twitter cyberbullying. The main difference with previous research is that we not only developed a machine learning model to detect cyberbullying content but also implemented it on particular locations real-time tweets using Twitter API. The entire approach to detect and prevent Twitter cyberbullying is divided into 2 major stages: developing the model and experimental setup. The Natural Language Toolkit (NLTK) is used for the preprocessing of data. NLTK is used for tokenization of text patterns, to remove stop words from the text, etc. The entire model is divided into 3 major steps: Preprocessing, the algorithm, and feature extraction. The first step in the solution is to collect the tweets from Twitter using

Twitter API. In the next two steps are data preprocessing and feature extraction is performed over the tweets. And after performing preprocessing and feature extraction tweets are passed to the SVM model for classification to predict whether the tweet is Bullying or Non-Bullying. The second step In this step, the proposed model has transformed the data in a suitable form which is passed to the machine learning algorithms. The TFDIF vectorizer [1] is used to extract the features of the given data. Features of the data are extracted and put them in a list of features. Also, the polarity (i.e. the text is Bullying or Non-Bullying) of each text is extracted and stored in the list of features. The third step To detect social media bullying automatically, supervised Binary classification machine learning algorithms like SVM with linear kernel and Naive Bayes is used. The reason behind this is both SVM and Naive Bayes calculate the probabilities for each class (i.e. probabilities of Bullying and Non-Bullying tweets). Both SVM and NB algorithms are used for the classification of the two-cluster.

**ADVANTAGES OF PROPOSED SYSTEM:**

* Naive Bayes models are used recommendation systems, sentiment analysis, and spam filtering. Naive Bayes algorithms are very easy to implement.
* After performing preprocessing and feature extraction on the dataset, for training and testing, and divided the dataset into ratios 0.45 and 0.55 respectively. Both SVM and Naive Bayes are evaluated to calculate the accuracy, recall, f-score, and precision. Interestingly SVM outperformed Naive Bayes in every aspect.
* the accuracies of both the Naive Bayes and SVM. The Support Vector Machine achieved the highest accuracy i.e. 71.25%, while Naive Bayes achieved 52.70% accuracy.

**Algorithms:** machine learning; classifiers; Naive Bayes; support vector machine (SVM); Twitter API.

**SYSTEM REQUIREMENTS:**

**HARDWARE REQUIREMENTS:**

* System : Intel Core i7.
* Hard Disk : 1TB.
* Monitor : 15’’ LED
* Input Devices : Keyboard, Mouse
* Ram : 16 GB.

**SOFTWARE REQUIREMENTS:**

* Operating system : Windows 11.
* Coding Language : Python
* Tool : PyCharm, Visual Studio Code
* Database : SQLite

**REFERENCES:**

**Date of Conference:**13-15 May 2020

**Date Added to IEEE *Xplore*:**19 June 2020

**ISBN Information:**

**Electronic ISBN:**978-1-7281-4876-2

**DVD ISBN:**978-1-7281-4875-5

**Print on Demand(PoD) ISBN:**978-1-7281-4877-9

**INSPEC Accession Number:**19711153

**DOI:**[10.1109/ICICCS48265.2020.9120893](https://doi.org/10.1109/ICICCS48265.2020.9120893)

**Publisher:**IEEE

**Conference Location:**Madurai, India