ASSESSMENT OF DIVERSE TECHNIQUES FOR CYBER BULLYING DETECTION ON TWITTER USING SENTIMENT ANALYSIS

Supervisor: Ms. K. Rohini (Assistant Professor)

Submitted by

K. Himabindu-16JG1A0563

P. Ramya Sree-16JG1A0589

P. Amrutha Manasa-16JG1A0584

P. Ruby Indu Meena-16JG1A0582

Department of Computer Science and Engineering, GVPCEW.

ABSTRACT

Cyberbullying in this era, is no more a new word. The word cyberbullying is defined as an aggressive act that is performed by an individual or in a group of people, using electronic forms of devices, repeatedly left with no chance to defend herself from the perpetrator. This bullying creates memories that last for life long. Even hearing the name of the person who bullied them, after years is enough to send the chills up on the backs of many people. Cyberbullying includes posting, sending, sharing the harmful or negative messages, false content about someone else. It can also include personal or private information about someone else which causes humiliation or embarrassment. In parallel, with endemic use of social media,

Cyberbullying is becoming more prevalent. In this paper, we discuss about the several stages data collection, pre-processing, applying various classifiers to predict the level of cyberbullying. Our main moto is to compare different sentimental analysis approaches to detect the cyberbullying using three machine learning algorithms. We give an output comparing these three algorithms out of which gives a highest accuracy in order to decide how to detect the cyberbullying activity on internet and to control the level of cyber risk in both the real and the virtual world.

KEYWORDS

Cyberbullying, Sentiment Analysis, Machine Learning algorithms (such as Support Vector Machine, Random Forest, Naïve Bayes), Social Media network like Twitter.

I. INTRODUCTION

During the last decades, the ways of communication had changed to a great extent.Even though, technology offers various advantages, it also has several side effects; for instance, web cams, online chats, texting, e-mail, instant messaging apps and many other websites might be used for hurting other people. In fact, the repeatable act of showing their aggressiveness intentionally over an indefensible victim via electronic means is known as cyberbullying.

With the recent advancement of social media, people adopted the new ways to spread their hate speech through various sites like Facebook, Twitter, Instagram and many which finally lead to cybercrime. Social Media has become the voice to many people to express their views and even their hate towards others.

Sentiment analysis in general defined as a technique that identifies whether the sentence is positive, negative or neutral. In other words, with the advancement of technology, sentiment analysis is used for text analysis, it is also known as emotion AI.

We believe that the first step to prevent cyberbullying is to identify the false content in the text through text analysis and this process is done through sentiment analysis. To evaluate the proposed stages, we extract the datasets from a popular microblog where people feel free to express their views or opinions and also address other users i.e., Twitter. We then apply data pre-processing, lemmatization, and then apply polarity to words that are identified as bullying and non-bullying words, finally we split the data into train and test dataset. To these datasets we apply the machine learning algorithms like SVM, Random Forest, Naïve Bayes and check the accuracy rates. Our results, in general shows the finest algorithm in order to detect the cyberbullying activity.

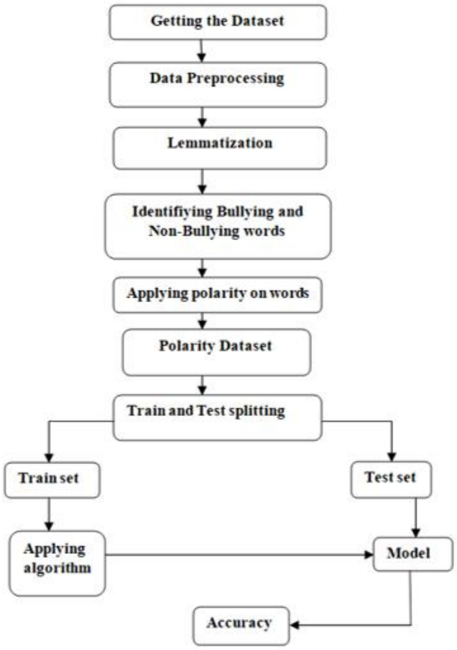


Fig. 1. Proposed Cyberbullying Detection Framework

The rest of paper is systemized as follows. Section II represents Dataset, Section III represents Proposed Method, Section IV represents Experiments and Section V represents Conclusion and last Section VI represents References.

II.DATASET

The dataset used is the tweets which are extracted from Twitter and these tweets are further divided into two different classes i.e., bullying and non-bullying words. We used Twitter API to extract the tweets from the Twitter, and then we applied data pre-processing where cleaning of data is done by removing the stop words, and then we apply lemmatization to retrieve root words which helps in splitting data into bullying and non-bullying words.

We collected the dataset of nearly 27000 tweets where we split the dataset into 80% of training dataset and 20% of testing dataset, which nearly out of 27019 tweets, 21,616 tweets are considered as training dataset and 5,403 tweets as testing dataset. In general, 80:20 is the basic ratio used for practice, it's quite rare to use the 50:50 ratio.

TABLE 1: The Total Dataset divided in the ratio of 80:20

Total Dataset	Training Set	Test Set
27019	21616	5403

III. PROPOSED METHOD

The method proposed in this paper includes data collection, feature extraction, classification and algorithms. The overall process is explained in detail in this section.

A. DATA COLLECTION

As mentioned, we collected the dataset from the social networking site i.e., Twitter. To get access and to gather the required dataset we needed the Twitter API for python which need an authorization key. For this, we need to create a developer account, so that we can get our keys and tokens from which we further collect our tweets.

Each tweet in dataset is segmented into bullying and non-bullying text. These tweets are collected using particular keywords such as depress, harassment, bullying, violence, suicide and many which prone to indicate bullying. Using Twitter API, we collected a corpus of dataset which is divided into two classes: positive tweets and negative tweets. These two types of collected data is used to train the classifiers to recognize the positive and negative sentiments.

Overall, we collected dataset nearly 27,000 tweets in which, we further divided the training and testing data into 80:20 ratio.

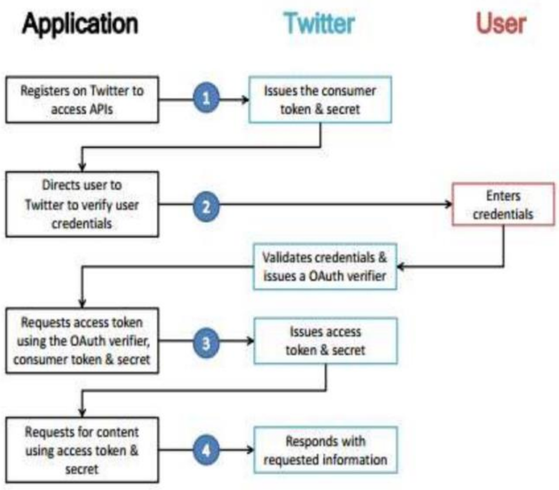

Application
Twitter
User
Registers on Twitter to access APIs
1
Issues the consumer token & secret
Directs user to Twitter to verify user credentials
2
Enters credentials
Validates credentials & issues a OAuth verifier
Requests access token using the OAuth verifier, consumer token & secret
3
Issues access token & secret
Requests for content using access token & secret
4
Responds with requested information

Fig. 2. Data collection using Twitter API

PRE-PROCESSING

Initially, for pre-processing we rectified the spelling mistakes in the tweets as most of the people tend to write in short forms such as: "ur so hot", and such type of tweets are converted to general English forms as "you are so hot". Then we convert all the uppercase letters into lower case letters followed by removing all the URLs and unnecessary white spaces.

Stop words are the words generally considered as the most commonly used words such as "is", "to", "a", "an" which the search engines are programmed to ignore. In this case, stop words can cause problems when searching for the phrases that include them, such as "The Who", "take that", "do you". Thus, we remove these stop words and also the URLs.

Further, we used the process lemmatization to which reduces the inflection in words to their root words such as mapping a group of words to the same stem. Thus, we complete the pre-processing.

FOLDING

From overall dataset we collected, we used 80% of our dataset as training dataset and 20% of it as testing dataset. Its is quite rare to 50-50 as folding ratio. From the 27,019 tweets, 21,616 tweets are considered as training dataset and 5,403 tweets are considered as testing the polarity of the tweets against the classifiers.

B. FEATURE EXTRACTION

I. **Stop Words:** Stop words are the words that are most commonly used such as- "a", "is", etc., which do not indicate any sentiment and such words can be removed. Thus, we filtered them out for feature extraction.

II. **Conversion of cases:** Converting the uppercase letters into lower case letters.

III. **Removal of URLs:** We remove all the URLs in the tweets.

IV. **Punctuation:** We remove all the punctuation marks such as- single quotes, double quotes, commas, question marks and so on.

As we process, we used TextBlob which is a python library and offers a simple API to access its methods and perform basic NLP tasks. We used sentiment function of TextBlob which in returns gives us the polarity of the tweets. If the polarity of the tweet is greater than 0 then it is considered as positive sentiment otherwise, it is considered as negative sentiment.

C. CLASSIFICATION

For detecting cyberbullying, we implemented various machine learning approaches in order to find the most efficient one. All of these techniques need training set which were collected, pre-processed and had been run through the classifier techniques. We implemented the three supervised algorithms, namely Support Vector Machine(SVM), Random Forest and Naïve Bayes to compare which each other to check how the results differ from one another.

COMPARISON OF MACHINE LEARNING TECHNIQUES

I. SUPPORT VECTOR MACHINE

In classification, the items are represented by their features, and these features need to be extracted from sample datasets. The datasets here consist of a database of tweets and hence, nothing serves better as the features other than the words in this document. Using TF-IDF(TermFrequency-InverseDocumentFrequency) vectorizer, we vectorized the input tweets into a format that the machine can identify. TF-IDF vectorizer also provides an efficient weighting scheme that makes it ideal for our situation. The feature weights for the training data is obtained, and this vocabulary is used in determining the feature weights of the test data as well, which is nothing but training our system for the test set of data. Scikit-Learn also provides for efficient calculation of accuracy and precision measurements.

In addition, the classifiers were adjusted using several parameters suchas the gamma value, Cvalue,etc.,.Gamma value reiterates the importance and influence of a single entry from the training set whereas C is the penalty value of error term. Too common words were ignored as it should not be classified as a feature of course and words too rarely appearing in our datasets are also ignored for they may appear only under special circumstances and might not affect polarity of the tweets by any means.

II. RANDOM FOREST

Random Forest Classifier is used which is a predefined method, having parameters n_estimaters, random_state. The parameter n_estimator is the number of trees to be used in the forest. Since, RandomForest is an ensemble method comprising of creating multiple decision trees.This parameter is used to control the number of trees to be used in the process. The default value of n_estimators can be changed from 0to100.

The parameter random_state, as the name suggests,is used for initializing the internal random number generator, which will decide the splitting of data into train and test indices. Setting random_state a fixed value will guarantee that the same sequence of the random numbers are generated each time you run the code. And unless there is some other randomness present in the process, the results will be same always. This helps in verifying the outputs.

In our paper, we used n_estimators as 200, thus the number of trees used in the for test are 200. And we mentioned the random_state to be 0 in fixed state, so that the same sequence of the random numbers are generated each time we run the code.

III. NAÏVE BAYES

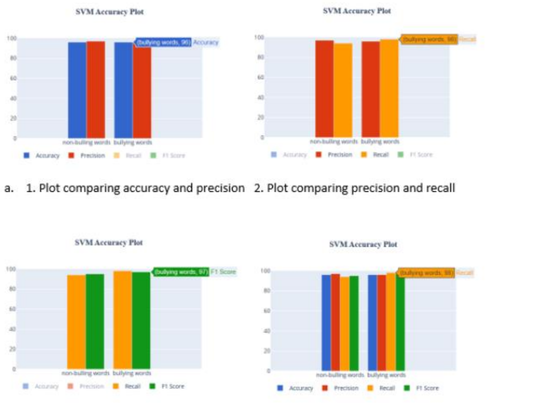
We implemented the python sklearn package for Naïvebayes classification. The training sets need to be labelled in order to recognize the category a corpusis classified upon. For example, if we are trying to find the gender mentioned in a particular corpus, the labels would be male and female. In our case, we are trying to detect bullying and hence we need to find out if a particular tweet is positive or negative. The negative tweets are regarded as cyberbullying related tweets. These labelled tweets are stored in a column that is added to the dataset. We collect a large dataset as mentioned, in order to get increase the accuracy rate.

We train the NaïveBayes classifier using the built-in package function with 21,616 tweets that are collected and annotated. Then we moved on to testing the polarity of the test data. In order to determine how much precise and accurate our classifier was, we also found out some metrics like precision, accuracy, recall and f1-score.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

SUPPORT VECTOR MACHINE

The linear kernel took considerably smaller time than RBF kernel, however it was still about 100 times slower than LinearSVC. The linear kernel is a wrapper of the python library libsvm. Precision values read 97% and 96% for positive and negative tweets respectively whereas the values for recall stands at 94% and 98% respectively.The accuracy of this classifieris found to be 96%.

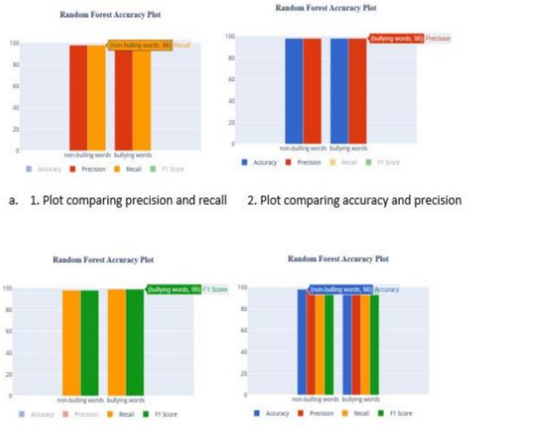


a. 1. Plot comparing accuracy and precision 2. Plot comparing precision and recall

b. 1. Plot comparing recall and f1-score 2. Comparision plot: accuracy,precision,recall,f1-score

Fig. 2. Resulting plots comparing accuracy, precision, recall and f1-score using SVM

RANDOM FOREST



a. 1. Plot comparing precision and recall 2. Plot comparing accuracy and precision

b. 1. Plot comparing recall and F1-score 2. Comparison plot: accuracy,precision,recall,f1-score

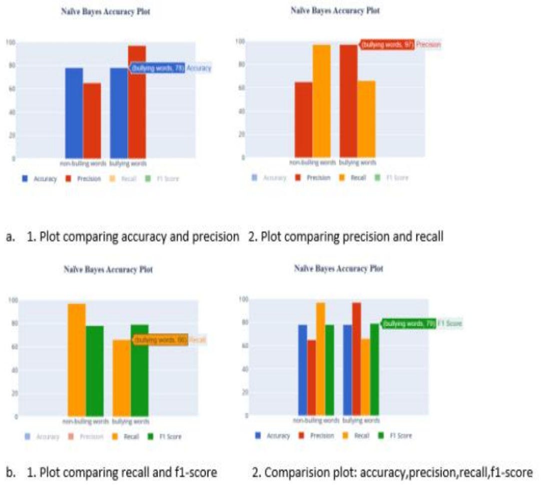
Fig. 3. Resulting plots comparing accuracy, precision, recall and f1-score using random forest

From the above plots, we can clearly say that we got the finest accuracy results. After training, a dataset consisting of large number of tweets were used to measure the metrics of the algorithm. After testing over 5,403 tweets an accuracy of 98% was obtained. The reason for this, is the classifier used.

NAÏVE BAYES

As shown above, the NaïveBayes Classifier provides of positive precision of 65% and recall of 97%. But these results are for positive tweets.One reason for such low positive precision and recall may be because the context of training tweets were mostly cyberbullying related, which means they had a lot of slang and hate words compared to nice and positive words.

We will focus more on the negative results since our context is cyberbullying. The negative precision came out to be 97% and recall was 66%. This means that most of its predicted sentiment was accurate when compared to its training set. Hence 78% of the negative tweets were relevant and 66% of the relevant negative tweets were retrieved. This means very few false positives were found for the negative class. However, many tweets that are negative are incorrectly classified. Low recall causes 44.16% false negatives for the negative label.



a. 1. Plot comparing accuracy and precision 2. Plot comparing precision and recall

b. 1. Plot comparing recall and f1-score 2. Comparision plot: accuracy,precision,recall,f1-score

Fig.4. Resulting plots comparing accuracy, precision, recall and f1-score using NaïveBayes

V. CONCLUSION

In this paper, we discussed how each machine learning algorithms performed in detection of cyberbullying from social media using sentiments. The negative tweets are defined as bullying tweets and hence we were able to detect successfully using the three machine learning approaches – among them, RandomForest stood the first best and Support Vector Machine performed as second best with minor changes. In this paper, we constructed a dataset containing the tweets of cyberbullying and proposed and evaluated a methodology for adequate classification of data. In addition, we explored the feasibility of this automatic cyberbullying detection. We have fine tuned our simulation by running tests again and again to show the best results and our analysis showed the reason for each of their performance.

VI. REFERENCES

[1] Moin Mostakim,M Sintaha, SB Satter, N Zawad, C Swarnaker and Ahanaf Hassan, “Cyberbullying Detection Using Sentiment Analysis in Social Media “,Institutional Repository, BRAC university, 18 August 2016

[2] Michele Di Capua, Emanuel Di Nardo,Alfredo Petrosino,"Unsupervised Cyber Bullying Detection in Social Networks",2016 23rd International Conference on Pattern Recognition (ICPR),Cancun center,Cancun Mexico, December 4-8,2016

[3]K. Nalini and L. Jaba Sheela,"Classification of Tweets Using Text Classifier to Detect Cyber Bullying",Springer International Publishing Switzerland 2015 S.C.Satapathy et Al, Emerging ICT for Bridging the Future-Volume 2, Advances in Intelligent Systems and Computing,338

[4]Amit Gupte, Sourabh Joshi,Pratik Gadgul,Akshay kadam,"Comparative Study of classification Algorithms used in Sentiment Analysis", International Journal of Computer Science and Information Technologies,vol.5(5),2014,6261-6264

[5]M.Ganesan and P.Mayilvahanan,"Cyber Crime Analysis in Social Media Using Data Mining Technique", International Journal of Pure and Applied Mathematics, volume 116, No.22 2017,413-424

[6]Mifta Sintaha,Moin Mostakim,"An Empirical Study and Analysis of the Machine Learning Algorithms Used in Detecting Cyber bullying in Social Media", 2018 21st International Conference of Computer and Information Technology (ICCIT),21-23 December,2018

[7]Suman Rani,Jaswinder Singh,"Sentiment Analysis of Tweets Using Support Vector Machine", Suman Rani et Al, International Journal of Computer Science and Mobile Applications,vol.5 Issue.10, October-2017,of 83-91

[8]Rekha Sugandhi,Anurag Pande,Siddhant Chawla, Abhishek Agarwal,Husen Bhagat,"Methods for Detection of Cyberbulling :A Survey",2015 15th International Conference on Intelligent Systems Design and Application (ISDA)

[9]Shylaja S S, Abhishek Narayanan,Abhijith Venugopal, Abhishek Prasad,"Document Embedding Generation for Cyber-Aggressive Comment Detection using Supervised Machine Learning Approach",14 International Conference on Natural Language Processing, December 2017,pages 348-355

[10]Krishna B.Kansara and Narendra M.shekokar,"A Framework for Cyberbulling Detection in Social Network",2015 International Journal of Current Engineering and Technology,vol 5,No.1,E-ISSN 2277-4106,P-ISSN 2347-5161

[11]Vinita Nahar,Sayan Unankard, Xue Li and Chaoyi Pang, "Sentiment Analysis for Effective Detection of Cyber Bullying", Springer -Verlag Berlin Heidelberg,2012,Q.Z.Sheng et Al,LNCS 7235,pages 767-774

[12]Maral Dadvar,Dolf Trieschnigg,Rowland Ordelman and Francisca Dr Jong,"Improving Cyberbulling Detection with User Context", Springer-Verlag Berlin Heidelberg,2013,P.Serdyukov et al ,LNCS 7814,pages 693-696