MODELS FOR AUTOMATIC DETECTION OF CYBERBULLYING ON ONLINE SOCIAL MEDIA

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

MASTER OF TECHNOLOGY
IN
COMPUTER SCIENCE & ENGINEERING

Submitted By-
ASIF AHMAD KHAN
2K20/CSE/07

Under the supervision of
Dr ARUNA BHAT
(Associate Professor)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

MAY, 2022

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road , Delhi-110042

DECLARATION

I Asif Ahmad Khan (2K20/CSE/07), student of M.Tech (Computer Science & Engineering), hereby declare that the project Dissertation titled "**MODELS FOR AUTOMATIC DETECTION OF CYBERBULLYING ON ONLINE SOCIAL MEDIA**" which is submitted by me to the Department of Computer Science & Engineering, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology is original and not copied from any source without proper citation. This work has not previously formed the basis for any Degree, Diploma Associateship, Fellowship, or other similar title or recognition.

Asif Khan

Place - Delhi
Date - 25 May, 2022

ASIF AHMAD KHAN
(STUDENT)

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road , Delhi-110042

CERTIFICATE

I, hereby certify that the project Dissertation titled "**MODELS FOR AUTOMATIC DETECTION OF CYBERBULLYING ON ONLINE SOCIAL MEDIA**" which is submitted by Asif Ahmad Khan (2K20/CSE/07), Department of Computer Science & Engineering, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of degree of Master of Technology in Computer Science and Engineering is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree to this University or elsewhere.

Place: Delhi
Date: 25 May, 2022

Dr ARUNA BHAT
(Associate Professor)

ABSTRACT

With time, use of the internet has become very common among people, and thus the rise of internet usage has given birth to a problem of cyberbullying. Cyberbullying can have a serious impact on the psychological health of the person who is the victim of it. Hence, detection of cyberbullying is required on the internet or social media. Much research has been done in the field of detection of cyberbullying. Machine learning is one of the approaches that can be used for automatic cyberbullying detection on online social media. Study on some of the papers related to cyberbullying along with some of the NLP techniques and different models used for cyberbullying detection tasks has been done . The graph, which is based on the papers reviewed, shows that the tf-idf is mostly used either directly or with a combination of other techniques for feature extraction in cyberbullying detection using machine learning.

Five different ML models are used for classification of tweet data as bully or non bully text. Performance calculations of the model are done using accuracy and confusion matrix. Tf-IDF feature extraction technique is used to convert text into vector form. Random Forest model performs best followed by LR model. All the accuracy of the models are shown graphically also.

ACKNOWLEDGEMENT

I am very thankful to my supervisor Dr Aruna Bhat (Associate Professor), Department of Computer Science & Engineering, Delhi technological University, Delhi from the bottom of my heart for her guidance and motivation. She inspired and guided me in completing this project.

I am also very grateful to all the faculty members of my department (Department of Computer Science & Engineering) for their support and guidance.

Place: Dehi

Date: 25 May, 2022

Asif Khan

ASIF AHMAD KHAN

(STUDENT)

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ML : Machine Learning

TF-IDF : Term Frequency -Inverse Document Frequency

BOW : Bag Of Words

WE : Word Embedding

RF : Random Forest

LR : Logistic Regression

KNN : K Nearest Neighbor

NB : Naive Bayes

DT : Decision Tree

CHAPTER 1: INTRODUCTION

1.1 OVERVIEW

Nowadays, the internet is easily accessible to all and most of us spend a large amount of time on the internet through online education, online gamings, e-commerce, social networking sites etc. With the increase of internet usage, the problem of cyberbullying also arises. Cyberbullying can be detrimental to a person's psychological health which can even make someone to commit suicide. Hence, monitoring is required for cyberbullying detection on the Internet/Social media. Many works have been done for the automatic cyberbullying detection using different approaches. One of the methods for the automatic cyberbullying detection is by using supervised machine learning techniques. Patterns used by bullies in their languages should be detected for cyberbullying detection tasks, and ML learning can be useful for this pattern detection [11]. In this supervised learning technique, we will have a dataset that is already labeled as bully or non_bully text, which will be further used for training various classification models. Once the training of an ML model with a given dataset is completed, the model is ready to predict the result of given text.

Using automatic detection of cyberbullying, binary classification can be done on text from social media to classify whether it is bully text or not. Multiclass classification is also performed in cyberbullying detection., Talpur, B.A., et al [10] performed multi class classification based on severity of cyberbullying . A dataset comprising a set of text data can not be directly used for a classification task. First, we need to convert this text into an n-dimensional input vector and this input vector can be given to different models for the classification of text. This text can be converted into input vectors using various NLP techniques like BOW, TF-IDF etc . To develop a successful ML model for detecting cyberbullying behaviour, various factors play a role, and the features used in the task of cyberbullying detection are the most important factor [10]. In order to make our ML model recognize and classify our text for the cyberbullying detection task, text is preprocessed and useful information is analysed from it [1].

1.2 RESEARCH QUESTION

1) What are the various models used for Automatic cyberbullying detection on text data.
2) What are the various featurization techniques used for Automatic cyberbullying detection ML models.
3) Which models perform better in classification of bully and non bully text.

1.3 OBJECTIVE

1) To perform a study on various models used for cyberbullying detection on text data.
2) To find out the various featurization techniques used for cyberbullying detection which uses ML models.
3) To implement a different ML model for cyberbullying detection on a dataset containing text data.
4) To find the best model for cyberbullying detection task on a given dataset by comparing the performance of all models.

CHAPTER 2: BACKGROUND

2.1 FEATURE EXTRACTION TECHNIQUE

Text data can not be given directly as an input to classification models, we need to convert these text into vector format, in order to accomplish this conversion from text to vector we need feature extraction techniques. If the two texts are similar or closer to each other, then their vectors should also be geometrically closer to each other, and if the texts are not closer or dissimilar, then their vectors should not be geometrically closer to each other. Such properties should be posed by n-dimensional input vectors. Some of the techniques used for feature extraction in cyberbullying detection tasks are,
Talpur, B.A., et al [10] introduced a new input feature and combined it with twitter API features and predicted features (gender,age etc) to create a n dimensional input vector. Hani , J., et al [11] created an input vector by adding sentiment analysis feature with tf-idf, Muneer, A., et al[13] uses tf-idf and word2Vec to get n dimensional input vectors from text. Shah, R., et al [9] uses tf-idf technique for the feature extraction. Lepe-Faúndez, M., et al [14] uses various ways to create input vectors for model using lexicon approach , combining lexicon with tf-idf, combining lexicon with word embedding , combining lexicon with tf-idf and word embedding . Ali, W. N. H. W., et al[15] uses n-grams bag of words technique for getting features from text.

2.1.1 Bag of Words (BOW)

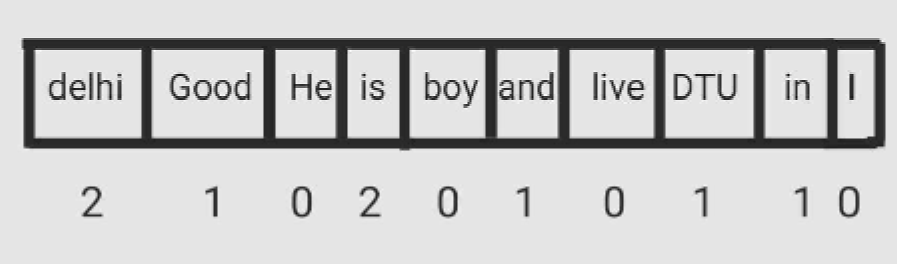
In this technique we count the occurrence of words in text, Let say we have three text

1) Text1: Good boy.
2) Text2: DTU is in delhi and delhi is good.
3) Text3: He lives in delhi.

In order to convert these texts into vector using BOW

Step1 : create a group of unique words from all the text.

Step 2: Now to convert text into vectors we will count the occurrence of unique words in a text. The vector for Text2 “**DTU is in delhi and delhi is good**”, is 2102010110 as shown below. Delhi occurs 2 times in text2, good occurs 1 time, He occurs 0 time and so on.



delhi	Good	He	is	boy	and	live	DTU	in	I
2	1	0	2	0	1	0	1	1	0

Fig 1. Vector representation of text using BOW

Generally a group of unique words to form vocabulary are large and text contains a limited number of words , the vectors formed by BOW will be sparse in nature.

2.1.2 TF-IDF

It is the multiplication of **term frequency** with an **inverse document frequency** of the word. Term frequency(tf) of a word is number of time word occur in documents divided by total number of words in a documents and inverse document frequency(idf) of a word is log of number of documents divided by number of documents containing that words.

Let say we have three documents

1) Doc1: Good weather
2) Doc2: Good morning
3) Doc3: morning is pleasant

To form a vector for document 1 , (tf * idf) will be calculated for a group of unique words in all documents.

tf(good)= (number of times **good** occurs in doc1)/(number of word in doc1) = ½ idf(good) = log(number of document / number of document containing word **good**)

= log(3/2).

tf(morning) = (number of times **morning** occurs in doc1) / (number of words in doc1).

= 0/2 = 0

So while creating a vector for doc1, tf value will be 0 for all the other words which do not contain in doc1, so the final vector of doc1 will have non zeros value for words containing in doc1 only.

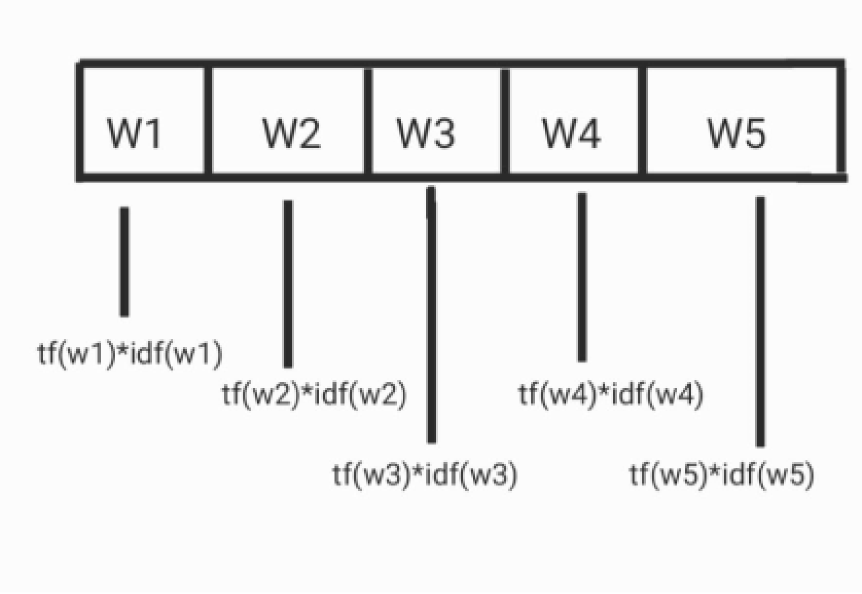


Fig 2. Vector representation of text using tf-idf.

2.2 CLASSIFICATION MODELS

Classification models are used to classify the data into various classes. Some of the classification models used for cyberbullying detection are Ali, W. N. H. W., et al[15] uses SVC Linear and Decision Tree model, Talpur, B.A., et al [10] uses Naive Bayes , SVM with RBF Kernel, Random Forest, and KNN model , Shah, R., et al [9] uses Multinomial NB, Random Forest, Logistic Regression, SVC, and SGD model, Hani , J., et al [11] uses Neural network and SVM model.

2.2.1 KNN Model

KNN is a classification algorithm in ML. In KNN model we find k nearest data points from our query point. The class of the query point will be the class that belongs to the majority of points among k points.

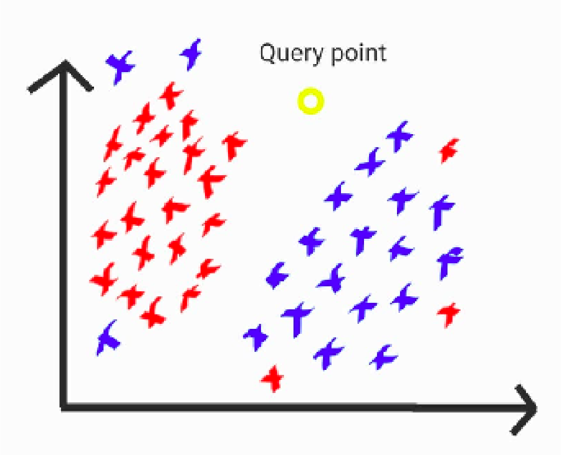


Fig. 3 KNN model

Above fig 3. Shows two classes of points one is red and other is blue, now we have a query point and we want to find which class the query point belongs to, let suppose k=5 then we will calculate the distance of data points from this query point and find 5 nearest data points. Out of 5 let suppose 3 points belong to blue class then because majority of points belong to blue class then our query point will also belong to blue class.

2.2.2 Random Forest Model

Random Forest[7] is an ML algorithm based on the technique of supervised learning. RF uses an ensemble method in which more than one classifier is combined. RF uses multiple Decision

Trees on a different subset of the dataset. All these DT models predict a result and based on majority voting final output is given.

2.2.3 Naive Bayes Model

NB is a classification algorithm in ML based on the concept of probabilities. In order to make a classification of a point, probabilities of all the classes are calculated and whichever class has the highest probability then the point belongs to that class.

CHAPTER 3: RELATED WORK

3.1 LITERATURE REVIEW

Alotaibi, M., et. al [1] proposed the automatic cyberbullying detection method by combining the features of 3 different models of deep learning and these 3 models are transformer block, CNN, and BiGRU. The proposed method classifies Twitter comments as offensive or not offensive. The authors combined the three known datasets and then the performance of the proposed method was measured. Seventy five percent of the data is selected as training data and the remaining data is selected as test data. The proposed model gives an accuracy of 87.99%. Apart from accuracy, the proposed method is also evaluated on four other different performance metrics.

Khairy, M. , et al [2] presented a survey work on the Arabic content for the automatic cyberbullying detection and abusive language. Authors analyse 27 studies on the contents which are in Arabic , among which 10 are on cyberbullying detection and 17 are on detection of the language which are offensive. In contrast to the definition of cyberbullying as a recurrent behaviour, all of the datasets used in the cyberbullying detection method were labelled with a one post. Most of the datasets used are imbalanced, which has an impact on the classifier's performance.

Rosa, H. , et al [3] did a detailed review of twenty two studies on automatic detection of cyberbullying and an experiment to validate current practices using feature engineering and two datasets. The authors use a quantitative systematic review approach for the automatic detection of cyberbullying.

Mozafari, M., et al [4] introduce a novel approach of transfer learning based on BERT to improve the detection of hate speech system's performance. BERT is an existing pre-trained language model . This model addresses the issue such as insufficient amount of labelled data of hate speech. 2 datasets which are available publicly are used by authors. Some biases while collecting the dataset can be detected by the model is observed in the result.

Al-Ajlan, M.A. , et. al [6] proposed a novelty algorithm CNN-CB for cyberbullying detection using convolutional neural networks. Proposed algorithm does not require feature engineering in detection of Cyberbullying. Algorithm uses word embedding concept and it is performing better than traditional approaches for Cyberbullying detection task.
Perera, A. , et al [8] presented a cyberbullying detection solution to find cyberbullying precisely along with the themes/categories related to cyberbullying using natural language processing (NLP) and supervised Machine Learning. Logistic Regression and SVM classifier is used in this cyberbullying detection system. Besides Tf/Idf, n gram and profanity along with sentiment analysis improves the system's accuracy. Accuracy of the solution proposed by Authors is 74.50%. Sarcasm text is not detected as cyberbullying in this proposed solution.

Shah, R., et al [9] presented a cyberbullying detection system on the Twitter dataset. The distribution of the dataset as non bully and bully text is equal. The Tf-idf method is used for feature extraction. Different classification models are used for cyberbullying detection. As a result, the authors found that Logistic Regression performed best among all classification models, with an accuracy of 93% and precision of 91%.

Talpur, B.A., et al [10] developed a cyberbullying detection feature-based machine learning model. Authors introduced a technique to create a new input feature. Along with the new input feature, other predicted features and Twitter API features were used. The results of the model are classified into multiple classes of non-cyberbullied, low, medium, and high levels. The model gives an accuracy of 93%.

Hani , J., et al [11] proposed an approach for cyberbullying detection using ML. For extracting the feature they have used sentiment analysis algorithm and tf-idf method. Classification tasks are evaluated using NN (neural network) and SVM classifiers on different n gram language models. Neural networks perform better than SVM. The accuracy of SVM with 4-gram is 90.3% and the accuracy of a neural network with 3-gram is 92.8%. The size of training data is limited for detecting patterns in cyberbullying, so to further enhance the performance of the model, a larger data size is required.

M. Ahmed, et al [12] performed sentiment analysis on data from Twitter. Authors use 3 ML models to classify the sentiment of a tweet into 5 categories.

Muneer,A., et al[13] has performed a comparative study of the model for cyberbullying detection with a global dataset compiled with unique tweets from Twitter. Performance is compared using seven machine learning models. Authors observe that performance of logistic regression classifiers improve with the increase of data size .For extracting the feature, tf/idf and word2vec are used. Among the seven classifiers, logistic regression performed best on the compiled global data set. The F1 score of the LR model is 0.9280, and its accuracy is 90.57%.

Lepe-Faúndez, M., et al [14] proposed different models using hybrid approaches that combine lexicons and machine learning for detection of aggressiveness in Spanish language. Five distinct ways to construct different models are proposed, each with its own way of extracting features from text. As a result, a hybrid model that uses lexicons provides the best results in the 3 language corpora of Spanish when compared to a model which does not use lexicons.

Ali, W. N. H. W., et al[15] proposed a model based on machine learning for cyberbullying detection using techniques like hyperparameter optimization , resampling and feature selection. SVC Linear and Decision Tree are used .Word-n grams technique is used for feature extraction. Eight various experiment setting were done to test the classifier , experiment setting like classifier + smote +feature selection , classifier + hyperparameter optimization etc. When tested using the x square test (feature selection) without any use of hyperparameter optimization and resampling, the Decision Tree classifier outperforms the SVC Linear classifier is shown in the result.

AlHarbi, B. Y., et al [17] presented a lexicon based approach for cyberbullying detection using sentiment analysis. PMI, Entropy, and Chi-square are the three different lexicon approaches used. A comparison is made among these three lexicon approaches to find which one is better for cyberbullying detection in Arabic text. Among all 3 lexicon approaches, the PMI approach gives the best results as compared to the remaining two approaches for cyberbullying detection in Arabic.

3.2 SUMMARIZATION OF RELATED WORK

Below Table 1. shows various models and methods used for Automatic cyberbullying detection tasks along with the results of these models. Apart from English text , the work on automatic cyberbullying detection was done in some other languages also.

Table 1. Various Models used for cyberbullying Detection

	Authors	*Models*	*Method Used*	*Result*	*Language*
1.	Alotaibi, M., et. al [1]	Transformer block , CNN and BiGRU.	Combining Features of 3 DL models.	Accuracy -87.99%	English

2.	Hani , J., et al [11]	SVM and Neural Network	Sentiment Analysis , N-grams and Tf-IDF method.	Accuracy (SVM) -90.3% and Accuracy (NN) -92.8%	English
3.	Talpur, B.A., et al [10]	NB, SVM with RBF Kernel, Random Forest, KNN.	New input feature , other predicted features and Twitter api features.	Accuracy -93%	English
4.	Shah, R., et al [9]	SVC, Multinomial NB, Logistic Regression, RF and SGD.	TF-IDF.	Accuracy (LR) -93%	English
5.	Lepe-Faúndez, M., et al[14]	22 different Model	5 Approaches for creating model Lexicon, WE_Lexicon, TF_IDF_Lexicon, WE_Lexicon_TF-IDF, and Ensemble approach.	Accuracies- Mexican corpus - 0.8431 Chilean corpus -0.892 Chilean-Mexican corpus- 0.8548	Spanish

6.	Muneer,A., et.al[13]	Seven Different ML Classifiers	TF-IDF and Word2Vec.	Accuracy (Logistic Regression) -90.57%	English
7.	Perera, A. , et al [8]	SVM and Logistic Regression	TF-IDF , N-gram, profanity, and sentiment analysis.	Accuracy -74.50%	English
8.	Al-Ajlan,M.A., et. al [6]	CNN-CB	Word Embedding	Accuracy -95%	English
9.	AlHarbi, B. Y., et al [17]	—	Sentiment Analysis and Lexicon Approaches PMI, Entropy and Chi-square.	Avg. F-score for PMI -81%	Arabic
10.	Ali, W. N. H. W., et al [15]	SVC Linear and Decision Tree	Techniques hyperparameter optimization , resampling and feature selection. N-grams, BoW.	Accuracy using default parameter (SVC Linear) -95.54%	English

The estimated numbers of some of the techniques used for featurization in the cyberbullying detection task based on the papers we reviewed are shown graphically in fig 4. The graph shows that tf- idf is used more frequently as compared to other techniques followed by WE(Word Embeddings) and N-grams in cyberbullying detection.

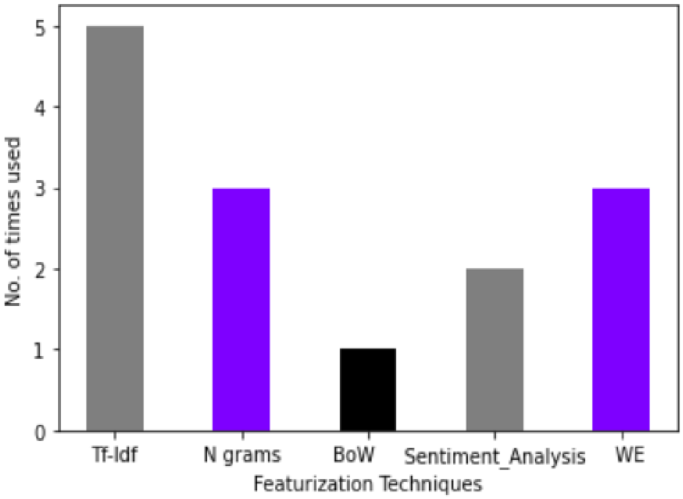


Fig 4. Graph for estimated no. of times featurization techniques used

CHAPTER 4: METHODOLOGY

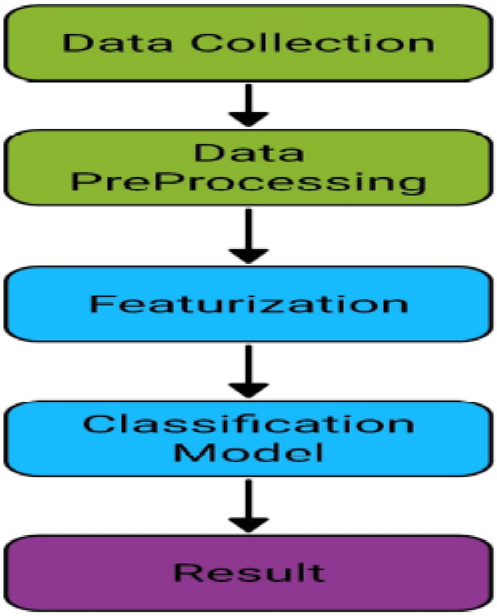
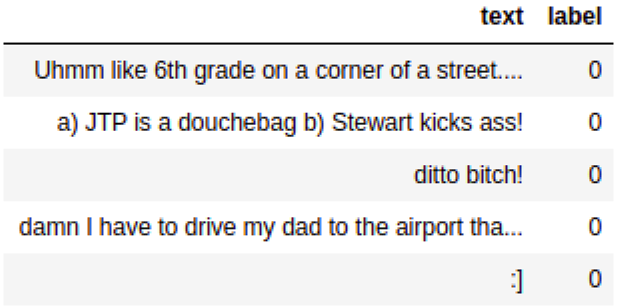


Fig 5. General steps for cyberbullying detection

4.1 DATASET

Dataset contains tweet data and the source of the data is from Kaggle[5]. Data is labeled as suspicious , cyberbullying (racism and sexism), hate and suicidal.

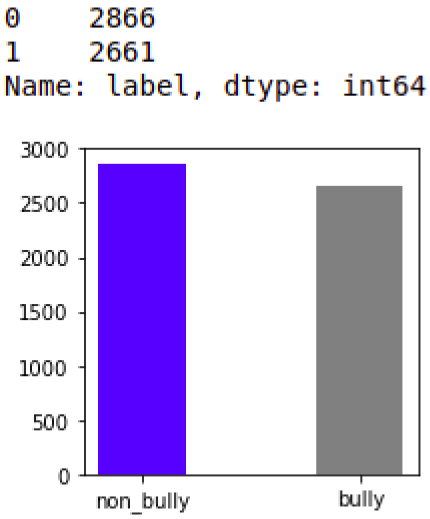
In order to perform binary classification for cyberbullying detection, data which is labeled as cyberbullying text is taken along with a sample of data which is not labeled as cyberbullying text. Bully text is labeled as 1 and non bully text is labeled as 0. Below figure 6. shows the sample of the dataset.



text	label
Uhmm like 6th grade on a corner of a street....	0
a) JTP is a douchebag b) Stewart kicks ass!	0
ditto bitch!	0
damn I have to drive my dad to the airport tha...	0
:]	0

Fig 6. Sample of dataset.

Below fig 7. is graphically showing the distribution of bully text and non bully text after cleaning and filtering the data.



```
0    2866
1    2661
Name: label, dtype: int64
```

Fig 7. Graph for distribution of data

4.2 DATA PREPROCESSING

In data preprocessing step text is processed by using various ways

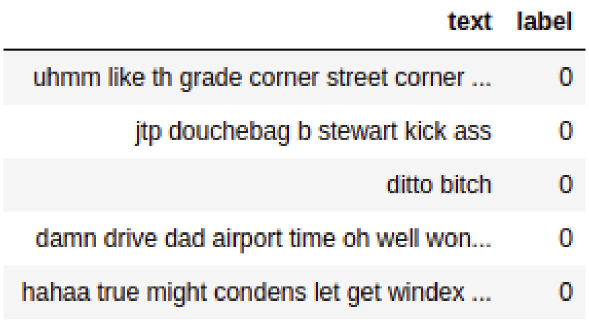
1) By removing any special symbols or character
2) By handling missing values
3) By removing url from text
4) Tokenizing the sentence into words
5) Converting the string into lower case
6) Removing the stopwords
7) Stemming or lemmatization etc.

Special characters and symbols can be removed by using regex in python. Example of removing special symbols using python library re is shown below.

text = re.sub(“ [special symbols] ”,” ”,text)

It will find these special symbols in a text and replace them with blank and this way special symbols will be removed from the text.

Various libraries can be used to remove stopwords from text, one of the ways to remove stopwords is by using nltk python library. Each word of a text is compared to a list of stopwords which can be obtained by using nltk python library , if a word contains in a list of stopwords, it will be removed from text. Finally stemming is performed on text using PorterStemmer which is imported from nltk library. Below figure 8. shows the sample of data which is preprocessed.

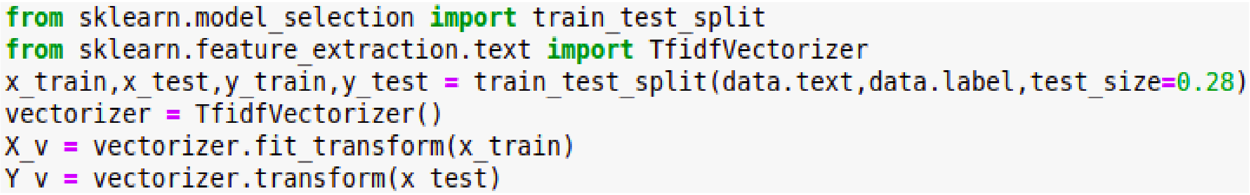


text	label
uhmm like th grade corner street corner ...	0
jtp douchebag b stewart kick ass	0
ditto bitch	0
damn drive dad airport time oh well won...	0
hahaa true might condens let get windex ...	0

Fig 8. Sample of Preprocessed data.

4.3 FEATURIZATION

TF-IDF featurization technique is used to convert text into n-dimensional input vector. TfidfVectorizer which is imported from sklearn python library is used to implement TF-IDF technique. Data is divided into training and testing data using train_test_split which is imported from sklearn python library. 28% of data is used as testing data and remaining data is used as training data. Training data x_train contains text data which is converted into vector form and stored in a X_v variable, Likewise x_test is testing data which contains text that is also converted into vector form and stored in a Y_v variable. Now it can be fed as input to various classification models. Tf-Idf technique for text to vector conversion is shown below in figure 9.



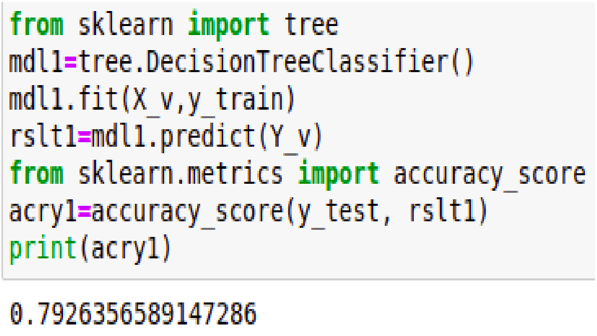
```
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
x_train,x_test,y_train,y_test = train_test_split(data.text,data.label,test_size=0.28)
vectorizer = TfidfVectorizer()
X_v = vectorizer.fit_transform(x_train)
Y v = vectorizer.transform(x test)
```

Fig. 9 Vector conversion using TF-IDF

4.4 TRAINING OF MODELS

The n-dimensional vectors are now given as input for training of classification models. The most crucial step in the architecture of text classification is choosing the best classifiers[10]. Various performance metrics such as accuracy, confusion matrix, log loss, F1-score, AUC, etc can be used to compare the effectiveness of different models for the automatic detection of cyberbullying tasks for a given dataset and can help to choose the best model. Five classification models which are used for classification of bullying text are Decision Tree , Random Forest , Logistic Regression , KNN and Naive Bayes. Sklearn python library is used to train our models with training data.

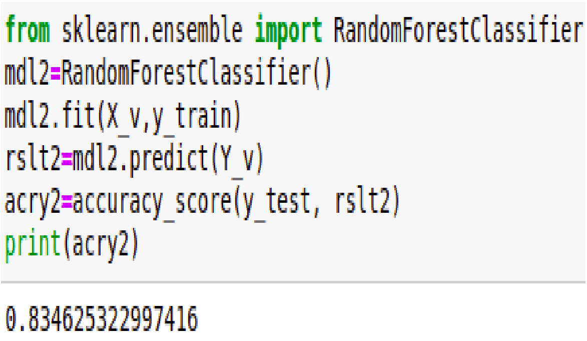
Below figures shows the code snippet for training of models using sklearn python library along with their accuracies on test data.



```
from sklearn import tree
mdl1=tree.DecisionTreeClassifier()
mdl1.fit(X_v,y_train)
rslt1=mdl1.predict(Y_v)
from sklearn.metrics import accuracy_score
acry1=accuracy_score(y_test, rslt1)
print(acry1)
```

```
0.7926356589147286
```

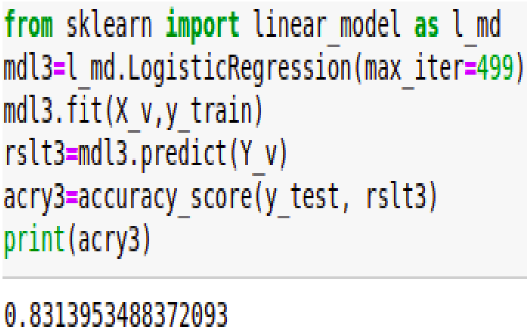
Fig 10. Training and accuracy calculation of DT



```
from sklearn.ensemble import RandomForestClassifier
mdl2=RandomForestClassifier()
mdl2.fit(X_v,y_train)
rslt2=mdl2.predict(Y_v)
acry2=accuracy_score(y_test, rslt2)
print(acry2)
```

```
0.834625322997416
```

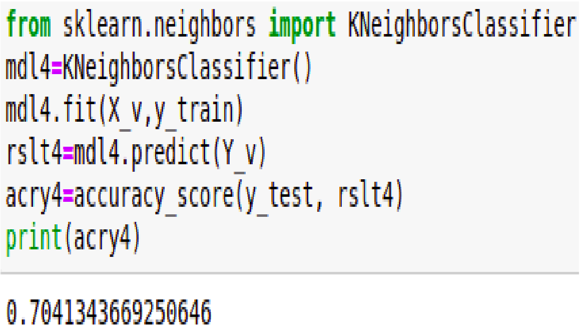
Fig 11. Training and accuracy calculation of RF



```
from sklearn import linear_model as l_md
mdl3=l_md.LogisticRegression(max_iter=499)
mdl3.fit(X_v,y_train)
rslt3=mdl3.predict(Y_v)
acry3=accuracy_score(y_test, rslt3)
print(acry3)
```

```
0.8313953488372093
```

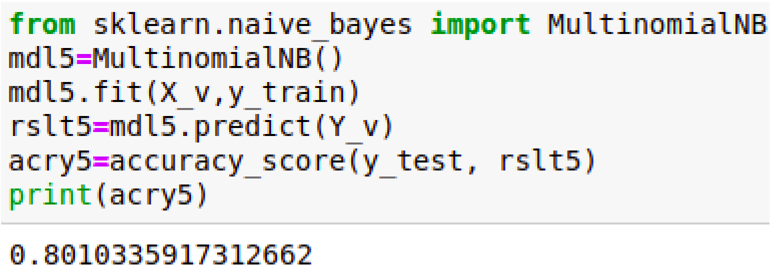
Fig 12. Training and accuracy calculation of LR



```
from sklearn.neighbors import KNeighborsClassifier
mdl4=KNeighborsClassifier()
mdl4.fit(X_v,y_train)
rslt4=mdl4.predict(Y_v)
acry4=accuracy_score(y_test, rslt4)
print(acry4)
```

```
0.7041343669250646
```

Fig 13. Training and accuracy calculation of KNN



```
from sklearn.naive_bayes import MultinomialNB
mdl5=MultinomialNB()
mdl5.fit(X_v,y_train)
rslt5=mdl5.predict(Y_v)
acry5=accuracy_score(y_test, rslt5)
print(acry5)
```

```
0.8010335917312662
```

Fig 14. Training and accuracy calculation of NB

CHAPTER 5 : RESULTS

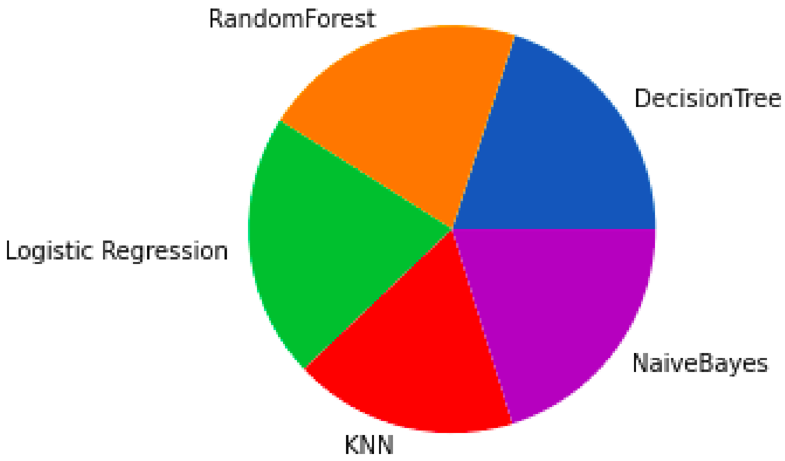


Fig 15. graph for accuracy of the models

Among all the classification models, random forest classifier is giving best accuracy followed by logistic regression for classification of bully and non bully texts. Accuracy and confusion matrices are used to measure the performance of models on test data. Below table 2. Shows the accuracy of all the models.

Table 2. Accuracy of the models

	DT	RF	LR	KNN	NB
Accuracy	79.26%	83.46%	83.13%	70.41%	80.10%

Below figures are showing the confusion matrix of all the five models.

```
[[652 155]
 [166 575]]

<AxesSubplot:>
```

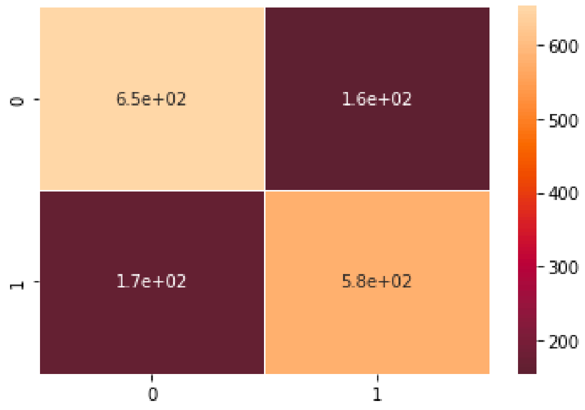


Fig 16. Confusion matrix for DT

```
[[729  78]
 [178 563]]

<AxesSubplot:>
```

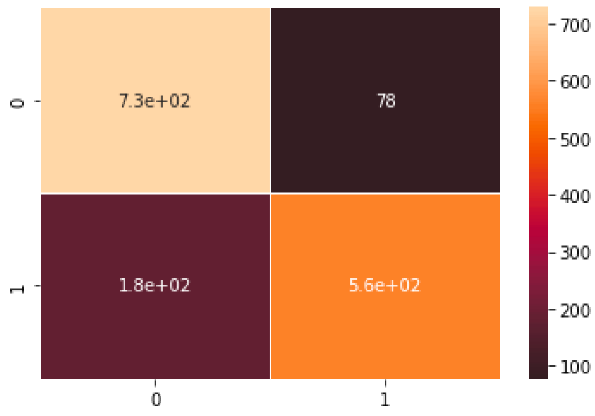


Fig 17. Confusion matrix for RF

```
[[718  89]
 [172 569]]

<AxesSubplot:>
```

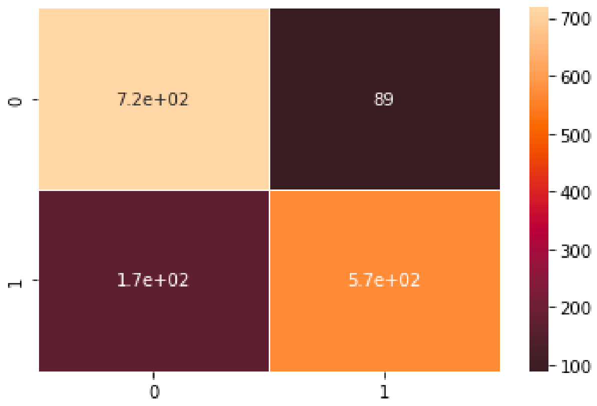


Fig 18. Confusion matrix for LR

```
[[712  95]
 [363 378]]

<AxesSubplot:>
```

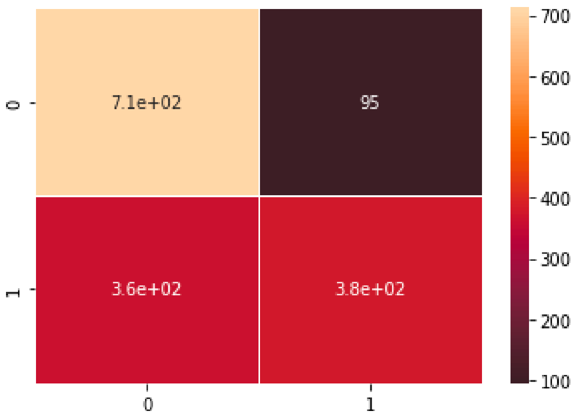
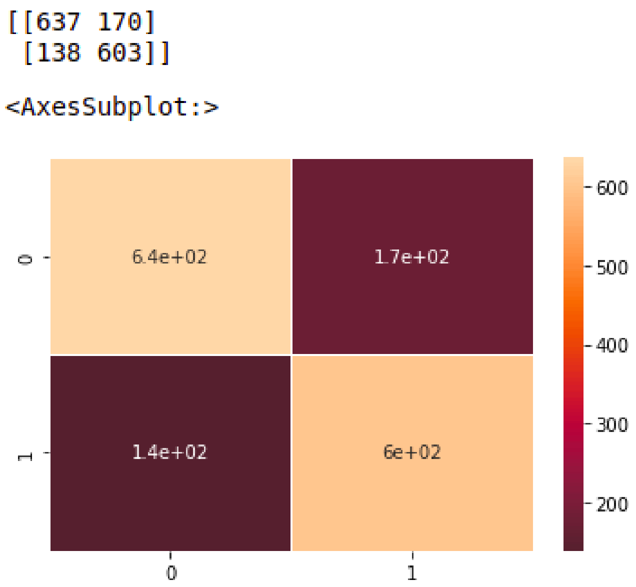


Fig 19. Confusion matrix for KNN



```
[[637 170]
 [138 603]]

<AxesSubplot:>
```

Fig 20. Confusion matrix for NB

CHAPTER 6: CONCLUSION

As we are all aware, the problem of cyberbullying over the use of the internet in our time is experienced by many internet/social media users. Cyberbullying is undesirable as it can have a bad impact on the psychological health of those who are bullied. Therefore, an automatic system for detection of cyberbullying should be implemented over the internet to control the problem of cyberbullying. Work has been done in the area of automatic detection of cyberbullying with the use of different feature extraction techniques, different ML and DL models, and even combinations of models. Some papers in the area of cyberbullying detection are reviewed. Different models and different methods are used for cyberbullying detection using machine learning. Tf- idf technique is mostly used with or without using another technique for getting input vectors from text in the task of cyberbullying detection, which uses an ML approach. Some studies of Automatic cyberbullying detection show that logistic regression classifiers perform better when compared to other ML classifiers.

Five different ML models are implemented for classification of bullying and non bullying text on a dataset consisting of tweets data. Random Forest model performs best with an accuracy of 83.46% followed by Logistic regression model with an accuracy of 83.13%. If the bullied text on social media is automatically detected on time, it may help in reducing the activities of cyberbullying on social network sites and can save many people from the negative effect of cyberbullying.

REFERENCES

[1] Alotaibi, M., Alotaibi, B., & Razaque, A. (2021). A multichannel deep learning framework for cyberbullying detection on social media. Electronics, 10(21), 2664.

[2] Khairy, M., Mahmoud, T. M., & Abd-El-Hafeez, T. (2021). Automatic Detection of Cyberbullying and Abusive Language in Arabic Content on Social Networks: A Survey. Procedia Computer Science, 189, 156-166.

[3] Rosa, H., Pereira, N., Ribeiro, R., Ferreira, P. C., Carvalho, J. P., Oliveira, S., ... & Trancoso, I. (2019). Automatic cyberbullying detection: A systematic review. Computers in Human Behavior, 93, 333-345.

[4] Mozafari, M., Farahbakhsh, R., & Crespi, N. (2019, December). A BERT-based transfer learning approach for hate speech detection in online social media. In International Conference on Complex Networks and Their Applications (pp. 928-940). Springer, Cham.

[5] Kaggle Dataset, https://www.kaggle.com/datasets/munkialbright/classified-tweets , May 2022.

[6] Al-Ajlan, M. A., & Ykhlef, M. (2018). Deep learning algorithm for cyberbullying detection. International Journal of Advanced Computer Science and Applications, 9(9), 199-205.

[7] RandomForest, https://www.javatpoint.com/machine-learning-random-forest-algorithm,May 2022.

[8] Perera, A., & Fernando, P. (2021). Accurate Cyberbullying Detection and Prevention on Social Media. Procedia Computer Science, 181, 605-611.

[9] Shah, R., Aparajit, S., Chopdekar, R., & Patil, R. (2020). Machine Learning based Approach for Detection of Cyberbullying Tweets. Int. J. Comput. Appl, 175(37), 51-56.

[10] Talpur, B. A., & O'Sullivan, D. (2020, December). Multi-Class Imbalance in Text Classification: A Feature Engineering Approach to Detect Cyberbullying in Twitter. In Informatics (Vol. 7, No. 4, p. 52). Multidisciplinary Digital Publishing Institute.

[11] Hani, J., Nashaat, M., Ahmed, M., Emad, Z., Amer, E., & Mohammed, A. (2019). Social media cyberbullying detection using machine learning. International Journal of Advanced Computer Science and Applications, 10(5), 703-707.

[12] M. Ahmed, M. Goel, R. Kumar and A. Bhat, "Sentiment Analysis on Twitter using Ordinal Regression," 2021 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON), 2021, pp. 1-4, doi: 10.1109/SMARTGENCON51891.2021.9645751.

[13] Muneer, A., & Fati, S. M. (2020). A comparative analysis of machine learning techniques for cyberbullying detection on twitter. Future Internet, 12(11), 187.

[14] Lepe-Faúndez, M., Segura-Navarrete, A., Vidal-Castro, C., Martínez-Araneda, C., & Rubio-Manzano, C. (2021). Detecting Aggressiveness in Tweets: A Hybrid Model for Detecting Cyberbullying in the Spanish Language. Applied Sciences, 11(22), 10706.

[15] Ali, W. N. H. W., Mohd, M., & Fauzi, F. (2018, November). Cyberbullying detection: an overview. In 2018 Cyber Resilience Conference (CRC) (pp. 1-3). IEEE.

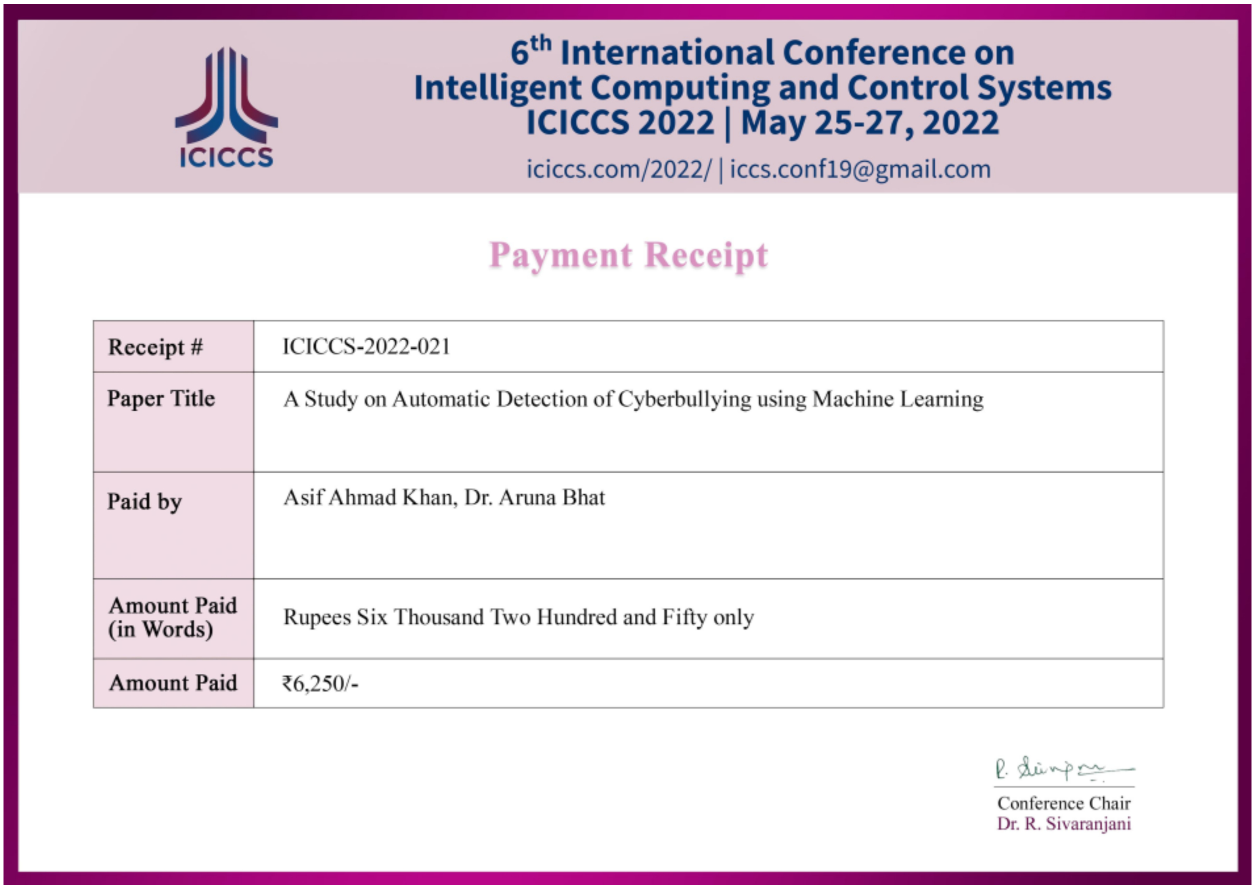
[16] R. Kumar and A. Bhat, "An Analysis On Sarcasm Detection Over Twitter During COVID-19," 2021 2nd International Conference for Emerging Technology (INCET), 2021, pp. 1-6, doi: 10.1109/INCET51464.2021.9456392.

[17] AlHarbi, B. Y., AlHarbi, M. S., AlZahrani, N. J., Alsheail, M. M., Alshobaili, J. F., & Ibrahim, D. M. (2019). Automatic cyber bullying detection in Arabic social media. Int. J. Eng. Res. Technol, 12(12), 2330-2335.

LIST OF PUBLICATIONS

ACCEPTED AND PRESENTED

[1] A. Khan and A. Bhat, **“A Study on Automatic Detection of Cyberbullying using Machine Learning”** in 6th International Conference on Intelligent Computing and Control Systems (ICICCS 2022) May 25-27, 2022 at Vaigai College of Engineering, Madurai, India.



ICICCS

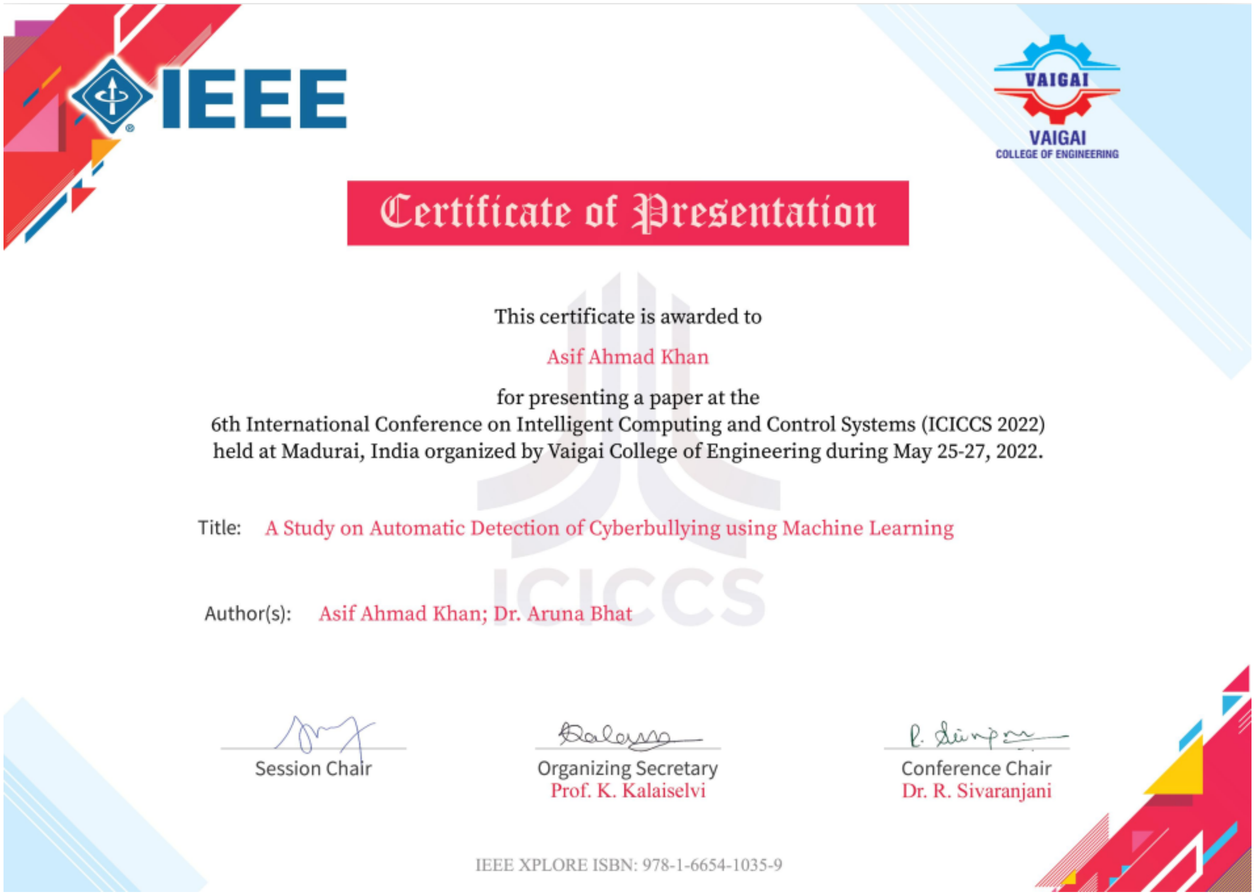
6th International Conference on Intelligent Computing and Control Systems ICICCS 2022 | May 25-27, 2022

iciccs.com/2022/ | iccs.conf19@gmail.com

Payment Receipt

Receipt #	ICICCS-2022-021
Paper Title	A Study on Automatic Detection of Cyberbullying using Machine Learning
Paid by	Asif Ahmad Khan, Dr. Aruna Bhat
Amount Paid (in Words)	Rupees Six Thousand Two Hundred and Fifty only
Amount Paid	₹6,250/-

Conference Chair
Dr. R. Sivaranjani



IEEE

VAIGAI
COLLEGE OF ENGINEERING

Certificate of Presentation

This certificate is awarded to

Asif Ahmad Khan

for presenting a paper at the
6th International Conference on Intelligent Computing and Control Systems (ICICCS 2022)
held at Madurai, India organized by Vaigai College of Engineering during May 25-27, 2022.

Title: A Study on Automatic Detection of Cyberbullying using Machine Learning

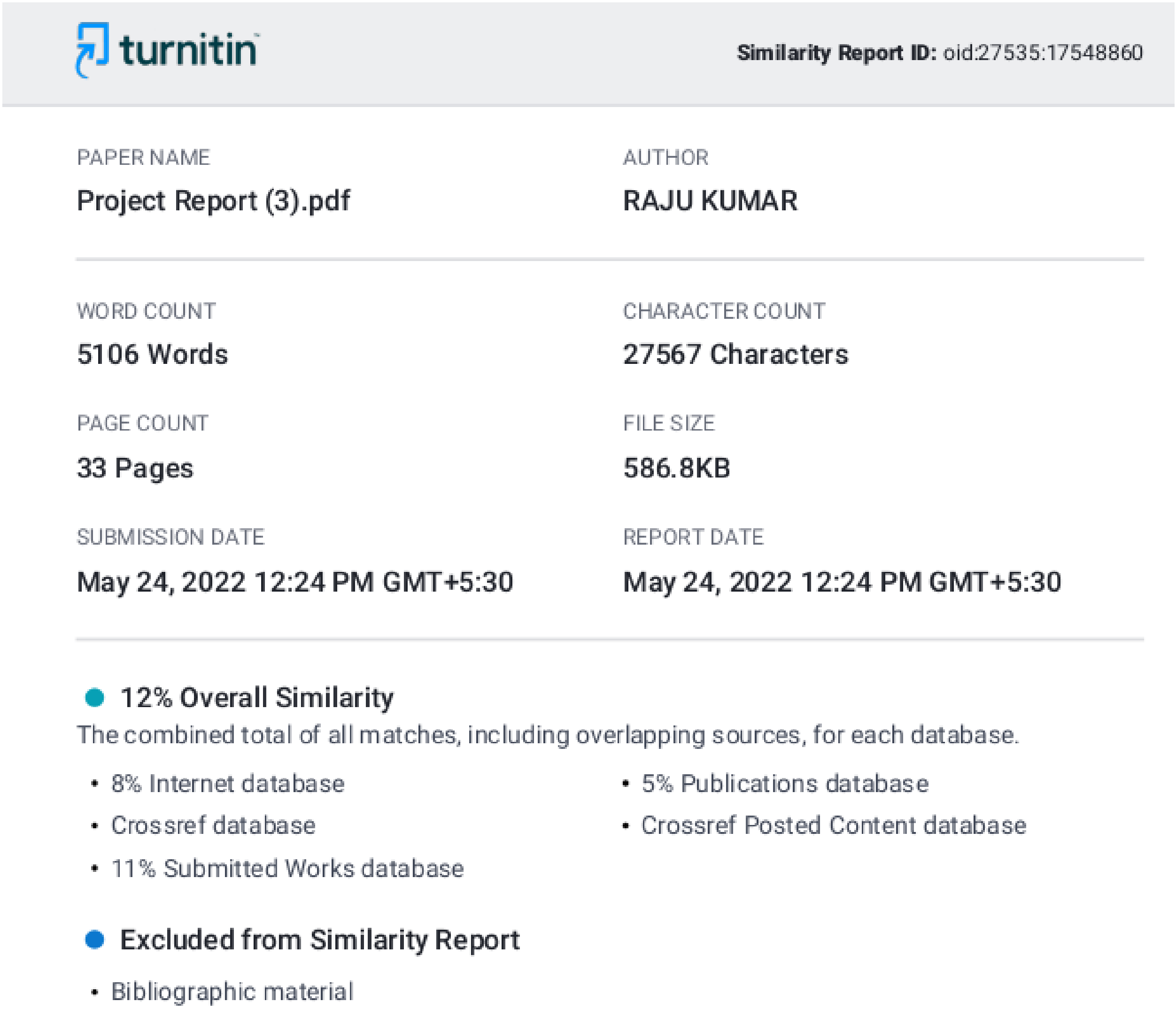
Author(s): Asif Ahmad Khan; Dr. Aruna Bhat

Session Chair

Organizing Secretary
Prof. K. Kalaiselvi

Conference Chair
Dr. R. Sivaranjani

IEEE XPLORE ISBN: 978-1-6654-1035-9



turnitin

Similarity Report ID: oid:27535:17548860

PAPER NAME

Project Report (3).pdf

AUTHOR

RAJU KUMAR

WORD COUNT

5106 Words

CHARACTER COUNT

27567 Characters

PAGE COUNT

33 Pages

FILE SIZE

586.8KB

SUBMISSION DATE

May 24, 2022 12:24 PM GMT+5:30

REPORT DATE

May 24, 2022 12:24 PM GMT+5:30

● 12% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 8% Internet database
- Crossref database
- 11% Submitted Works database
- 5% Publications database
- Crossref Posted Content database

● Excluded from Similarity Report

- Bibliographic material