Project Report On

Public Shaming Analyzer using Random Forest Classifier

By

Mr. Romin Katre

Mr. Chirag Narkar

Mr. Harsh Kore

Under Guidance of

Prof. Swati Verma

Department of Information Technology

Vidyavardhini's College of Engineering & Technology

University of Mumbai

2021-2022

Vidyavardhini's College of Engineering & Technology

Department of Information Technology

Certificate

This is to certify that the following students

Mr. Romin Katre

Mr. Chirag Narkar

Mr. Harsh Kore

have submitted project report entitled

Public Shaming Analyzer using Random Forest Classifier

as a part of their project-work in partial fulfillment of Semester VIII for the award of degree of ***Bachelor of Engineering*** *in* ***Information Technology*** *during academic year 2021-2022.*

Internal Guide : _____ ( )

External Guide : _____ ( )

Internal Examiner : _____ ( )

External Examiner : _____ ( )

Dr. Ashish Vanmali
HOD - IT,
VCET, Vasai

Dr. Harish Vankudre
Principal,
VCET, Vasai

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Romin Katre ( )

Chirag Narkar ( )

Harsh Kore ( )

Date : _____

Acknowledgment

We would like to express heartfelt gratitude to my internal guide, Prof. Swati Verma, for rendering all possible help and support for the implementation of the idea successfully. We would also like to thank our parents and friends that helped to complete the paper in a limited time frame. I am equally grateful to our faculty of management for their support. This paper includes collective efforts and dedication from our group members. The success and outcome required a lot of guidance from many people and we are very fortunate to have got all this help

Romin Katre
Chirag Narkar
Harsh Kore

Abstract

Public shaming in on-line social networks and associated online public boards like Twitter, Fb and Instagram have been growing in recent years. These occasions are acknowledged to personally have a devastating effect on the victim's social, political, and monetary life. Notwithstanding its acknowledged sick effects, little has been finished infamous online social media to treat this, regularly with the aid of using the excuse of giant extent and form of such feedback and, therefore, an unfeasible wide variety of human moderators required to acquire the project. In this research, we automate the project of public shaming detection in social media structures from the mindset of sufferers and discover ordinary aspects, namely, occasions and shamers. It is discovered that out of all of the collaborating customers who publish feedback in a completely specific shaming event, the bulk of them are able to disgrace the victim. Interestingly, it is also the shamers whose follower counts boom quicker than that of the non-shamers in several social media. Finally, primarily based totally on categorization and type of shaming tweets/feedback, a web utility has been designed and deployed for on-the-fly muting/blockading of shamers attacking a victim.

Contents

List of Figures

List of Abbreviations

API	Application Programming Interface
ML	Machine Learning
RF	Random Forest
IP	Internet Protocol Address
NLP	Natural Language Processing
PSA	Public Shaming Analyser

Chapter 1

Introduction

1.1 Overview

Everywhere, every human being has gone through an insult or a shame be it personally or on a public platform. Shaming includes abusing, discriminating based on color, race, gender, sexual preference, religion, political opinions, etc. This rapid growth in the number of people connecting worldwide, especially on public platforms has led to growth in the number of hate speech, shaming, and cyberbullying. A few examples are "Women are nothing more than a scumbag", "Shit Person you dont deserve to live". This increase in the number is nothing less than an alarm about how humankind has become insensitive. As a result, there is a need in detecting online shaming and automate the process to mitigate it. This field has been progressed and taken into consideration due to the grievousness of this shaming problem. This will help to know the aspect of shaming, the zone of remark, various visuals, periodic aspects of posting, targeted victim, a community of culprit, etc. This paper presents to you the contribution done for detecting the shaming done on social media platforms and taking actions against the user and making this whole process automated.

1.2 Problem Statement

Every individual is targeted online whether they have done something wrong or not. Even its normal human tendencies to have differences of opinion on a topic but are cursed, abused on the difference of opinion. One can hide their identity on various public platforms and can harass them to take any drastic step. [1] One can also influence an opinion that may lead to making wrong decisions sometimes. Moreover, for a

limited number of comments, you can personally delete/ block the particular comment. But comment sections for public figures have a multitudinous number of comments and dealing with it manually is something that we can say is a tedious job. The papers purpose is to learn how to make such a tedious job easy by using various Machine Learning algorithms by constructing a public shaming analyzer machine learning tool that gives you the list of comments which will classify is it abusive/ shaming or not.

The first extraction of the comments from a particular account will be done after going through the comment section of all the posts of a particular account, by using a machine trained by one of the algorithms of machine learning it will be classified as shaming or not

1.3 Motivation

India being a developing country and leading its way towards technology reformations we must also be ready to attack the dark side of technology. Cyberbully is one of the dark sides which has surfaced in the coming period of time and the after-goods are commodities everyone should be apprehensive of. Essentially women are the target of cyberbully. India recorded the loftiest position of online importance, with 45 of the repliers having endured cyberbully. Eight out of 10 people in India have endured some form of online importunity, with 41 of women have faced sexual importunity on the web, according to a new check commissioned by cybersecurity results establishment, Norton by Symantec. The most common forms of online importunity were planted to be abuse and cuts, which were reported by 63 repliers.

Chapter 2

Review of Literature

The paper researched by Dr. Andrew and Bridianee mentions the given points, on social networking sites, people tend to cross their boundaries having the misconception that no actions can be taken against them. They insult, abuse, and shame someone without thinking about the consequences. But according to the survey, actions were taken against them on a large scale which resulted in a loss of jobs or a losing position from a reputed committee. To achieve this progress has been made using different ways. Firstly, talking about the dataset, many datasets are available on opensource platforms. Waseem and Hovy released a public data set of 16,000 tweets labeled in one of the three categories: racist, sexist, or none in their research paper Hateful Symbols or Hateful People. They achieved an F1 score of 0.73 using character n-grams with logistic regression.

Similarly, work done by P Badiatiya provides an F1 score of 0.93 using deep neural networks on the same dataset. Manually annotations were done to bring them into the desired format. Word generalization, stemming and various other features are used to clean the dataset.

Adolescents minds tend to learn vulgar and explicit content quickly which may have an impact on their minds, to counter this the authors of "Detecting offensive language in social media to protect adolescent online safety" suggest using different methods to mitigate various shaming related issues on social sites, and further predict a users potentiality to send out offensive contents which are part of our research as well. Different approaches have been seen. Machine learning classifiers have been commonly used to classify into different categories like sarcasm, abuse, comparison, and many others.

From performance point of view, many people have tried with different types of hate speech dataset. One of the approaches have analysed that the performance of the model have almost no affect with respect to external factors like geographic or word length of the speech/comment but a major affect was seen when genders were considered. No.of hate speech/comment was increased when women was taken into consideration as compared to men.

Chapter 3

Report on The Present Investigation

3.1 Method

In this section, we present our procedure for making a public shaming analyzer with the highest accuracy possible. The procedure is divided into smaller steps to attain maximum accuracy, success, and is free of bugs. The main objective of this project is to analyze social media comments. The entire project is divided into three steps

1. Scraping of the comments

2. Training and Testing of the Machine Learning Model.

3. UI of the Project

1. Scraping of the comments The Primary Step of our project is to get real-time comments on a particular account. It was achieved by selenium for scraping real-time comments. Selenium helps to automate the testing of web applications. Additionally, we need a chrome driver to get access and control of the chrome web browser as the algorithm visits each post's URL, and then the comments are extracted.

2. Training and testing of the Machine Learning model The next and most important step of the project is to train and test the model. For Training our model we have used a random forest algorithm.For training there were many data set on open source, combining it and manually annotations were done to train from every aspect. Dataset

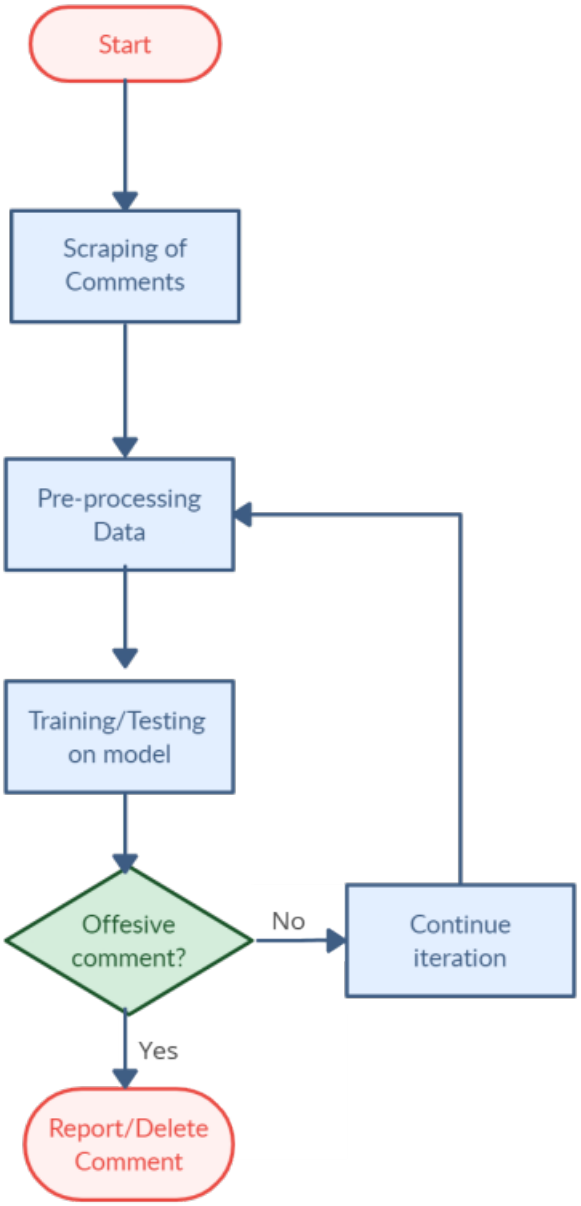

Figure 3.1: Procedure Flow

was split in an 80:20 ratio where 80 was used in training and 20 was used in testing.

As discussed in the overview the project is divided into three steps. This section gives information about the detailed work done

3.2 Scraping the comments

To get real-time comments this is a list of things that were used:

3.2.1 Selenium

For Automating our web app on browsers selenium is used. Selenium has different functions. For the research, we have used a selenium web driver.

3.2.2 Selenium- web driver

In simple terms, the selenium web driver automates and controls your browser. It helps you to connect your code to the site given in your command prompt and complete further actions. For automation of the browser, selenium, and selenium- web driver was used but for actually extracting we used our approach which extracts the comments from a particular profile using the page source of the currently open post in the browser. First, we studied the page source of a particular social media platform. Every account must have a post and every post must have comments. So after studying we observed that every single post's link is unique but written in the same format for example "instagram.com/p/CVbCJploIcc/", "https://www.instagram.com/p/CValA8hoqOU/". As it is seen two posts of the same account have links that have unique values (CVbCJploIcc, CValA8hoqOU) whereas the common thing is it starts with "/p/" so considering that we started splitting source code and took the link of every post. Now every comment has a unique ID so again splitting the source code we extracted the comments one by one.

3.3 Pre-processing, Training, and Testing of the model

3.3.1 Pre-processing

After getting the raw comments from a social media profile the next step is pre-processing. This step is important as the dataset on which the model is to be trained contains a lot of noise and elements which either don't play role in determining the

nature of the comment or are just merely an outlier, for example: “How is the life going in Seattle “here the stop words (‘is’, ‘the’, ‘in’) don’t hold weight hence can be ignored. This removal of “Stop Words” is further explained with other pre-processing techniques.

Stop-words

These are the words that are the most commonly used in a language. In English the stop words can be “the”, “us”, “our”, “in” etc. All these types of words need to be removed as we are working on text classification, these words don’t give us any information about the text and hence can’t be given to a model for training and restricting the unwanted data from our corpus.

Stemming

This is the process of producing the variants of a root word or reducing a word to its root word. Considering an example, the words “Likes”, “Liked”, “Likely” and “Liking” can be reduced to their root words which are “Like”. But this cannot be applied in every case as it is prone to being erroneous hence lemmatization is also employed.

Lemmatization

This approach is similar to stemming i.e., reducing the words to their root words, but lemmatization is much more widely applicable as while converting the words to their roots the context of the original words is also considered and taken care of. Let’s say we have the word “Caring” if stemming is used the root word comes out to be “Car” but if the lemmatization approach is used the output is changed to “care” and it is evident from here that the context of the original words is preserved.

Tokenization

It Is a process of splitting a sentence or paragraph into small units called tokens. This is an important step if we want to get the meaning of the sentence given, as the words present in the sentence gives us the meaning of the sentence rather than considering a whole sentence. For example, “Technology is Good” can be tokenized into [“Technology”, “is”, “Good”]. This helps us to determine the number of words in the sentence, it can also help us get information about the frequency of a particular word in the sentence. Numeric and Special character removal As we are doing text analysis, numeric values and special characters don’t play a major role in determining the meaning of the sentence hence has to be removed from the raw message.

Normalizing the slang

While dealing with the social media comments it is very obvious to have slang language as a part of a raw comment, but the model can't be trained on the slang words hence it becomes important to convert slang into original words so that they can provide us with the information which can be used for deciding future. For example, "I luv u" has to be converted to "I love you" so that it makes sense. To achieve this, we need a huge dictionary of slang words with their original words. Here is a small version of it slang dict='luv':'love','wud':'would','lyk':'like','wateva':'whatever','ttyl':'talk to you later', 'kul':'cool','fyn':'fine','omg':'oh mygod!','fam':'family','bruh':'brother','cud':'could', 'fud':'food'

Separating Hashtags

As we have comments as our inputs it is not surprising to have hashtags, we have to store hashtags separately as they generally hold weight in determining the context of a sentence as positive or negative.

Figure 3.2: Negative Words

Figure 3.3: Normal Words

3.3.2 Training and Testing

For training purposes, there are many classification algorithms among which we have used random forest algorithms as they gave high accuracy when it came to text analysis.

Random Forest Classifier

It is a machine learning classifier that is mostly known for regression and classification-related problems. It gives results to complex problems by combining many classifiers randomly. Decision trees are the bottom-up approach of random forest algorithms. The components of decision trees are root nodes, decision nodes, and leaf nodes. The dataset is categorized into different branches by the decision trees until it reaches the leaf node. There are several decision trees in a random forest.

Random forest is trained through bagging. The outcome of a random forest is based on the highest number of votes given. Different decision trees give different predictions and the prediction with the highest number is selected and given as output. In short Random Forest algorithm is known for giving a single result by taking the output of multiple decision trees. Important parameters that affect the output are node size the number of trees, and the number of features sampled. One of the benefits of the random forest algorithm is it reduces the risk of overfitting. In finance and healthcare, random forest is trusted because of the accuracy given. On testing this algorithm, we got an accuracy of 0.94(94 percent) with stemming and lemmatization and 0.83(83

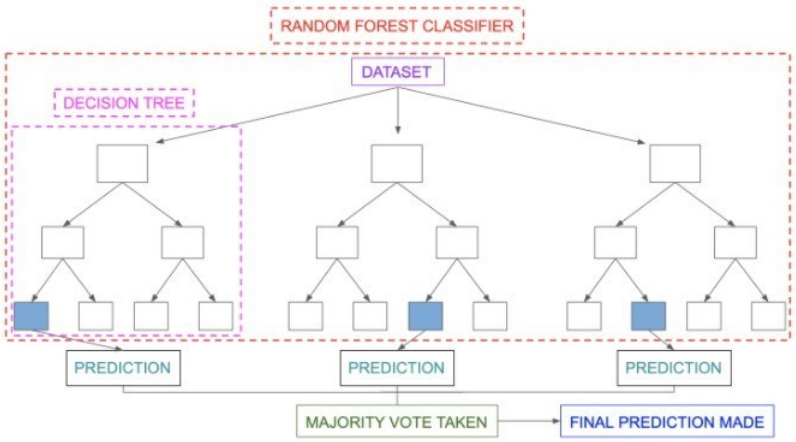

Figure 3.4: Random Forest Architecture

percent) without stemming and lemmatization below is the report attached in result sections.

3.4 UI of the Project

For UI we will be making a web application.For displaying our data on the web we will be using Streamlit.

3.4.1 Streamlit

It is an open source app for machine learning and data science projects . It is basically a framework in python which allows the users to create a web application with a simple and pure python script. Using a streamlit user can create an interactive web application without spending much time on the development of the app with the introduction of streamlit developing a dashboard for your machine learning solution has been made incredibly easy. We have many features provided by Streamlit that we can use to design our web page. We use the streamlit feature to create different functionalities like creating the header, sidebars and many more things. Once we are done with designing our page we can use our loaded model in order to make predictions In our project we have used two input boxes for user input. One is for username of the instagram and other one is for password. After submitting, the user input will be given to model for further process and output will be shown on the right hand side using status bar.

3.5 Project Algorithm

Step 1: START
Step 2: Login to user profile
Step 3: Access User Profile Data
Step 4: Scrape/Load User data
Step 5: Pre-processing Data
Step 6: Predicting comment type on Model
Step 6.1: If comment is predicted as a negative comment, unfollow/block user
Step 6.2: If comment is predicted as a positive comment, do nothing
Step 7: STOP

First of all, setting up the environment and python interpreter and loading all the libraries in the interpreter, installing if required, we initialize the streamlit client, as we have use streamlit to make the project available as a webapp hence this becomes a necessary step before doing anything after setting up the environment. Once the streamlit is up and running we get a local IP or a public IP depending on runtime use case. On accessing this IP, we get a login page which basically takes input as Username and Password. These login credentials are required to access the user's social media content.
Once logged in successfully, the next main step is to scrape the data using selenium and chrome webdriver. After studying the page source of Instagram, we can easily scrape the links for each and every post associated with that particular user profile and hence t gets easier to extract the comments for further analysis. The scraped data is stored in dictionary wherein the key is the username of the user who has commented and the value consist of all the comments from that user which will be used fr further analysis.
After extracting the data, pre-processing becomes the next most important step as we can't feed raw data to the model. In pre-processing we have implemented various filter as explained above namely: a. Stop words removal b. Stemming c. Lemmatization d. Tokenization e. Normalizing Slang f. Separating hashtags After cleaning the data this data is passed to the model for inferencing. The model is trained on similar dataset on Random Forest Classifier, this model gives a simple output as "1" or "0", where "1" denotes that the passed comment was a negative comment and consist of bad words, cyber bullying, body shaming etc. whereas "0" denotes that the comment is normal in nature and does not contain and type of negative content hence can be kept untouched. This model output is stored in dictionary associated with respective comment.
After getting the comment output aa "1" or "0" from the dictionary, further decisions

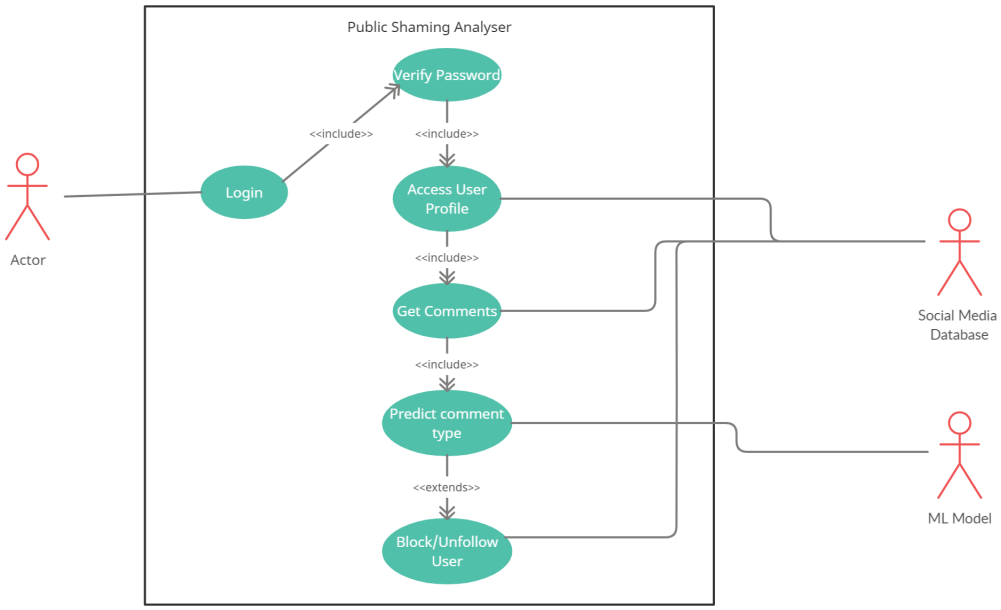

Figure 3.5: Use Case Diagram

are made whether or not to remove that user from commenting on your post again or do nothing. We have used Instagram API for tackling this problem, the API can be easily used and implemented, all we need to do is pass the username we want to unfollow and it will do the job just. But Instagram API has some restrictions as if used as free user then it only works for specific number of hits on that day.

Chapter 4

Results and Discussions

Result of implementing various steps, as it is impossible to cover every emotion of a comment. After implementation, these are results given on various steps of input.

Following are some of the implementation results.

Random Forest Classifier Report (Without Stemming and Lammetization)

Predicted	0	1
Actual		
0	3215	172
1	168	2120

	precision	recall	f1-score	support
0	0.80	0.85	0.84	3387
1	0.82	0.82	0.81	2288
accuracy			0.83	5675
macro avg	0.84	0.84	0.86	5675
weighted avg	0.84	0.84	0.86	5675

Figure 4.1: Output before implementing Stemming and Lemmatisation

```
...  Random Forest Classifier Report
     Predicted     0     1
     Actual
     0          3221   166
     1           166  2122
                   precision    recall  f1-score   support

                0       0.95      0.95      0.95      3387
                1       0.93      0.93      0.93      2288

         accuracy                           0.94      5675
        macro avg       0.94      0.94      0.94      5675
     weighted avg       0.94      0.94      0.94      5675
```

Figure 4.2: Output after implementing Stemming and Lemmatisation

Here it is clearly evident that after implementing Stemming and Lemmatisation we have gained better accuracy. The previous researches had an accuracy of 82.5%, whereas our model is performing with an accuracy of 94%

Figure 4.3: Actual comments from instagram profile

```
Post 1 :
 {'"#Roomateswag"', '"Nice"'}
```

Figure 4.4: Actual comments extracted from instagram profile

Figure 4.5: Actual comments from Facebook profile

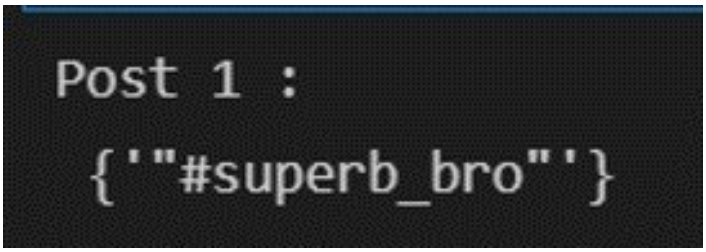

Figure 4.6: Actual comments extracted from Facebook profile

Figure 4.7: Output from real time extracted comment by the model

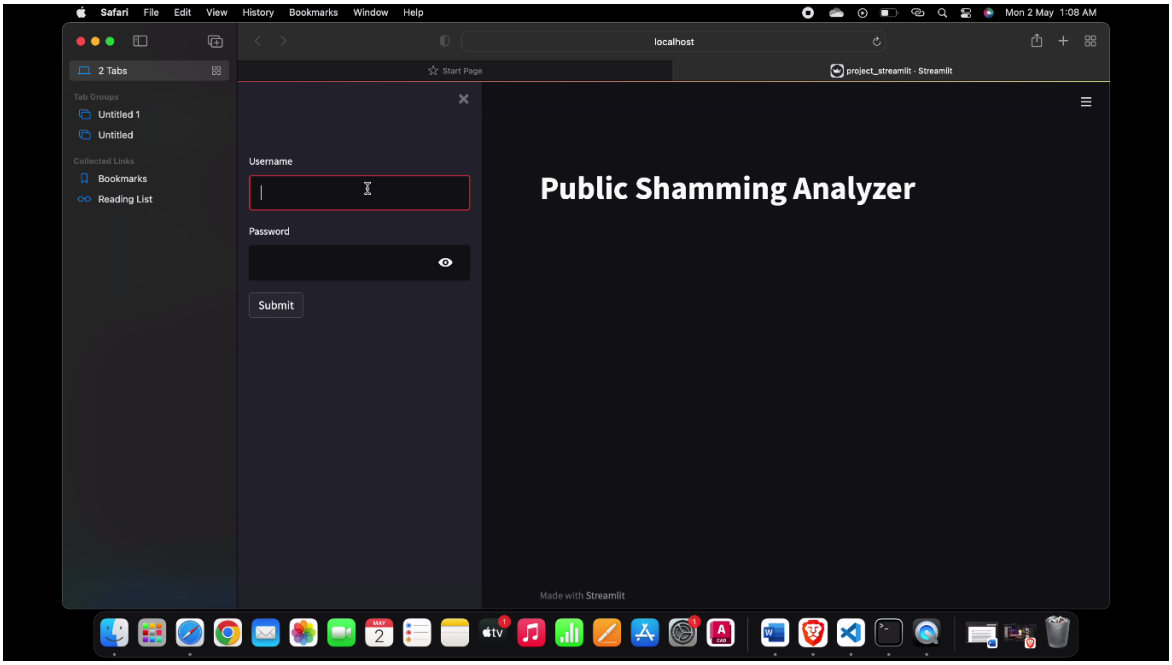

Figure 4.8: Beginning of the UI

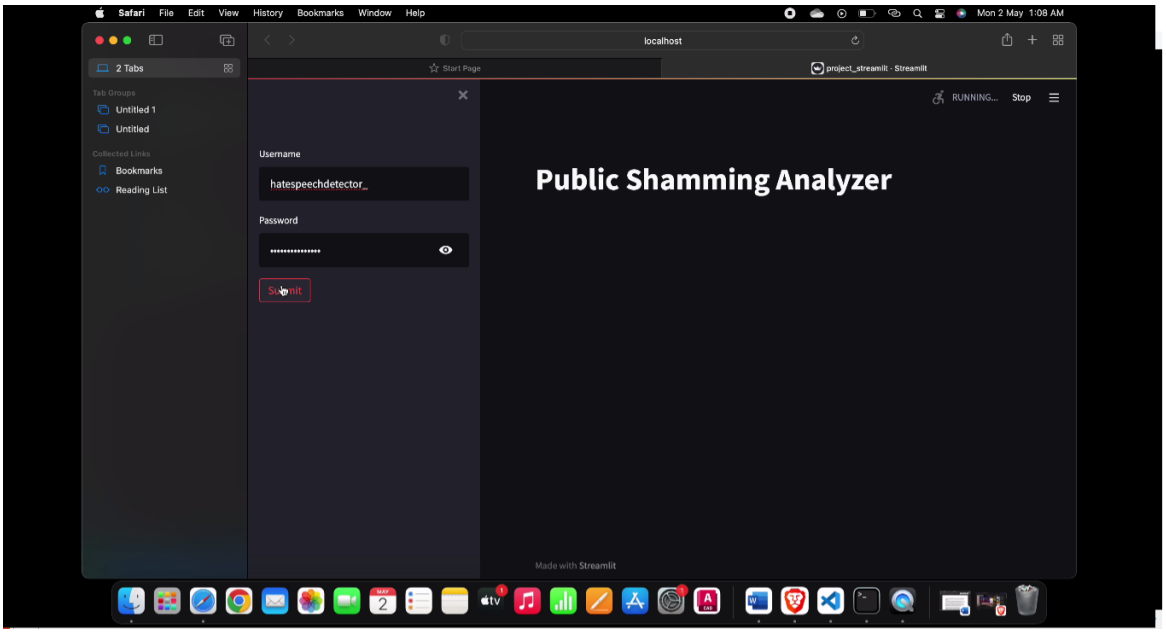

Figure 4.9: Giving User Input

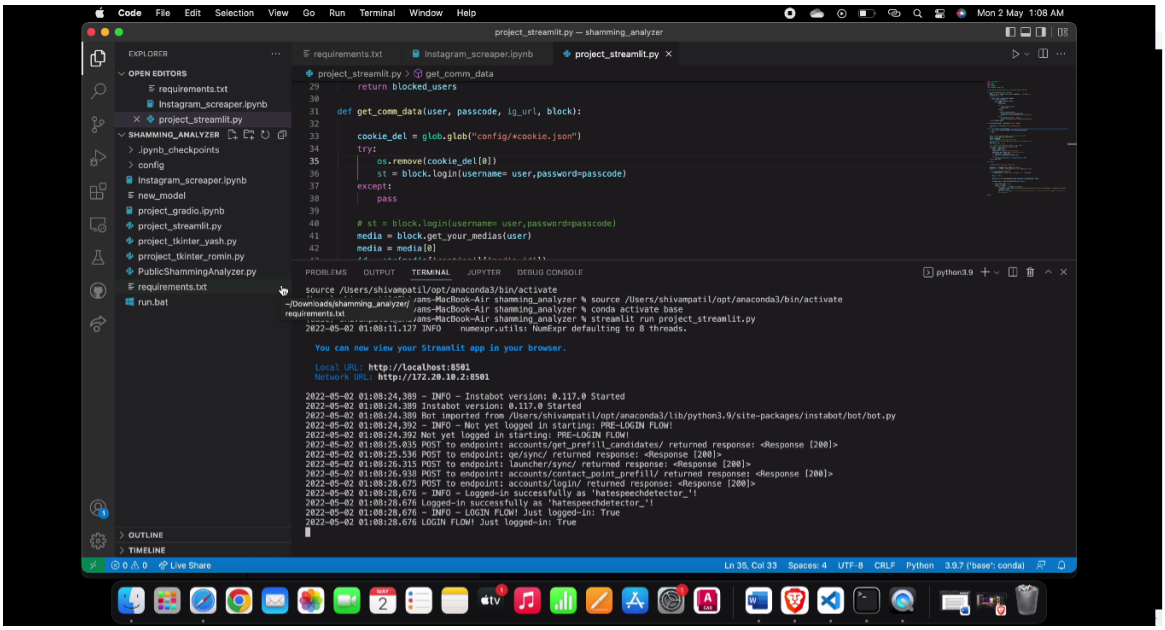

Figure 4.10: Login Executed

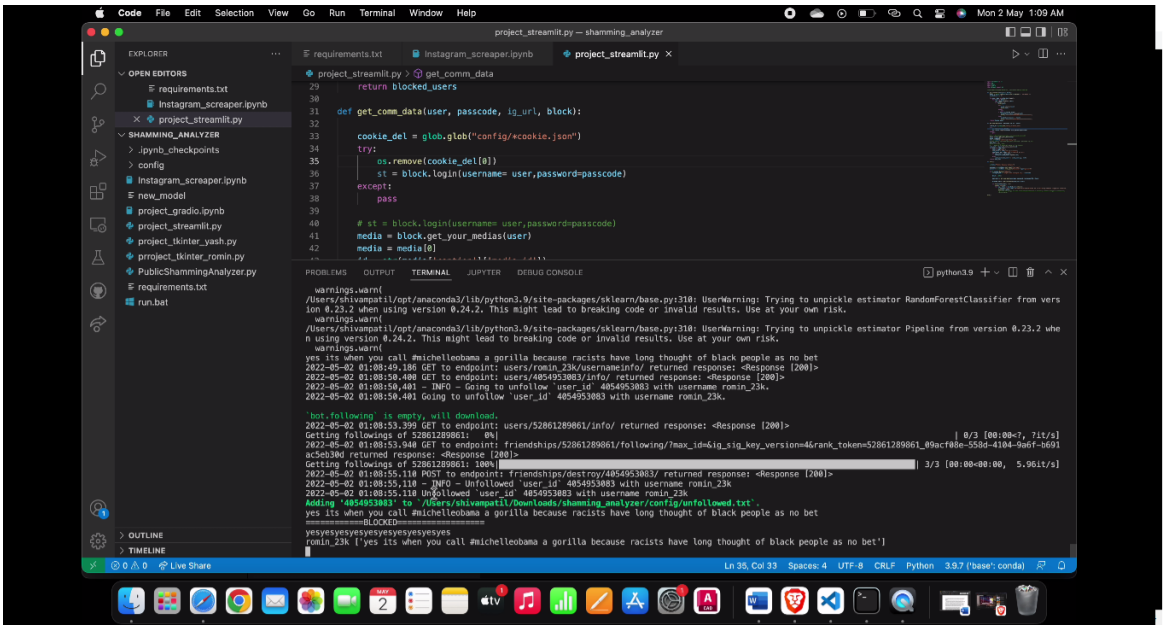

Figure 4.11: Unfollow/Block has been done on terminal

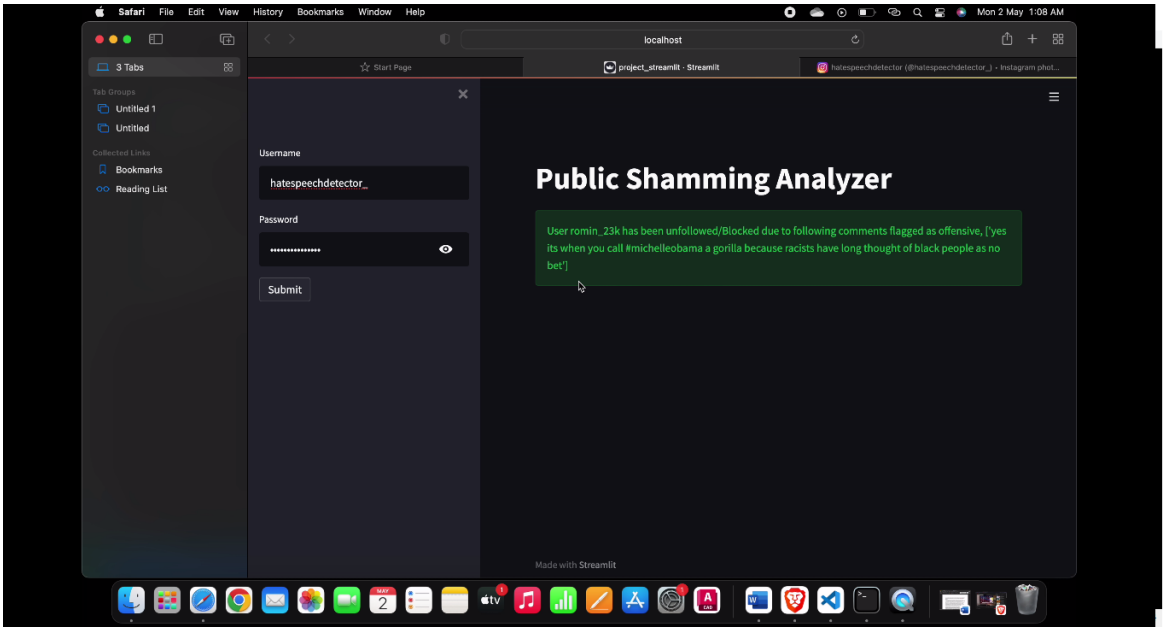

Figure 4.12: Final Output

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In recent days there are several studies have been done on text summarization but using text summarization for cyberbullying needs more attention as the growth of cyberbullying and hate speech on social media is a common case now. Analyzing the comments on a particular social media post can be helpful in a variety of cases as it can be used on a personal level for your social platform integrity or it can be used by a business, in general, to monitor their public image as it becomes important as well as difficult when it is a big business with a lot of public interaction. In this research, we have implemented a Random Forest algorithm for text analyzing the result we got without Stemming and Lemmatization is 0.83 (I.e., 83 percentage) which is good but the accuracy can be drastically increased using stemming and lemmatization which is 0.94 (i.e., 94 percentage). Hence it can be concluded that Random Forest can be a great approach in text summarization and can be used in the future for further work related to the same, it is likely that in near future, automatic cyberbullying detection can be excellent and useful as compared to manual checks.

5.2 Future Work

For future work, the current project only analyze the comments, but in the future, we can predict the personality of a user based on their comment section, the content posted, and their search section and this can be achieved using the deep learning method also the various method can be implemented for improving accuracy in random forest method.

We can also add easy API integration so that people can directly use the project and make extra additions to the current version, an easy API integration also helps people to directly integrate this project to their use case and ultimately make it easy to handle and implement

Appendix A

1. Python: is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java. The language provides constructs intended to enable writing clear programs on both a small and large scale. Python features a dynamic type system and automatic memory management and supports multiple programming paradigms, including object-oriented, imperative, functional programming, and procedural styles. It has a large and comprehensive standard library. Python interpreters are available for many operating systems, allowing Python code to run on a wide variety of systems. CPython, the reference implementation of Python, is open source software and has a community based development model, as do nearly all of its variant implementations. CPython is managed by the non-profit Python Software Foundation.

2. Natural Processing Language: Natural Language Processing (NLP) is a branch of computer science that includes aspects of human language and artificial intelligence. Machines utilise this technology to comprehend, analyse, manipulate, and interpret human languages. It aids developers in organising their information in order to execute tasks like translation, automated summarization, Named Entity Recognition (NER), audio recognition, relationship extraction, and topic segmentation.

3. Random Forest Classifier: It is a machine learning classifier that is mostly known for regression and classification-related problems. It gives results to complex problems by combining many classifiers randomly [9]. Decision trees are the bottom-up approach of random forest algorithms. The components of decision trees are root nodes, decision nodes, and leaf nodes. The dataset is categorized into different branches by the decision trees until it reaches the leaf node. There are several decision trees in a random forest.

Random forest is trained through bagging. The outcome of a random forest is based on the highest number of votes given. Different decision trees give different predictions and the prediction with the highest number is selected and given as output. In short Random Forest algorithm is known for giving a single result by taking the output of multiple decision trees. Important parameters that affect the output are node size the number of trees, and the number of features sampled. One of the benefits of the random forest algorithm is it reduces the risk of overfitting.

References

[1] Bridianee ODEA, Dr. Andrew CAMPBELL, "Online Social Networking and the Experience of Cyber-Bullying" Annual Review of Cybertherapy and Telemedicine 2012.

[2] Anna Schmidt, Michael Wiegand, "A Survey on Hate Speech Detection using Natural Language Processing" Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, Valencia, Spain, April 3-7, 2017.

[3] Rajesh Basak, Shamik Sural, Niloy Ganguly, Soumya K. Ghosh "Online Public Shaming on Twitter: Detection, Analysis, and Mitigation", IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, VOL. 6, NO. 2, APRIL 2019.

[4] Zeerak Waseem, Dirk Hovy "Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter," Proceedings of NAACL-HLT 2016, pages 88–93.

[5] Thomas Davidson, Dana Warmsley, Michael Macy, Ingmar Weber, "Automated Hate Speech Detection and the Problem of Offensive Language" arXiv:1703.04009v1 [cs.CL] 11 Mar 2017.

[6] Ying Chen, Sencun Zhu, Yilu Zhou, Heng Xu, Detecting offensive language in social media to protect adolescent online safety.

[7] Armand Joulin, Edouard Grave, Piotr Bojanowski, Tomas Mikolov, Bag of Tricks for Efficient Text Classification, Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics.

[8] Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, Yi Chang, "Abusive Language Detection in Online User Content", International World Wide Web Conference Committee, April 11–15, 2016, Montréal, Québec, Canada

[9] Gerard Biau, "Analysis of a Random Forests Model", Journal of Machine Learning Research 13 (2012) 1063-1095..

[10] Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, Vasudeva Varma. "Deep Learning for Hate Speech Detection in Tweets." 2017 International World Wide Web Conference Committee.

[11] Kristiawan Nugroho, Edy Noersasongko, Ahmad Zainul Fanani, Ruri Suko Basuki Improving Random Forest Method to Detect Hatespeech and Offensive Word , 2019 International Conference on Information and Communications Technology (ICOIACT).

[12] https://en.wikipedia.org/wiki/Random forest/media/File:Random forest diagram complete.png.

Publications and Awards

Publications

The following list of publications, presented at scientific conferences and published in reputed journals, contains work that is part of this report:

1. Proceedings of the 16th INDIACom; INDIACom-2022; IEEE Conference ID: 54597 2022 9th International Conference on "Computing for Sustainable Global Development", 23rd - 25th March, 2022 Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi (INDIA)

Proceedings of the 16th INDIACom; INDIACom-2022; IEEE Conference ID: 54597
2022 9th International Conference on "Computing for Sustainable Global Development", 23rd - 25th March, 2022
Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi (INDIA)

Public Shaming Analyzer using Random Forest Classifier

Romin Katre
Dept. of Information Technology
Vidyavardhini's College of Engineering & Technology
Mumbai, India
romin.182094105@vcet.edu.in

Chirag Narkar
Dept. of Information Technology
Vidyavardhini's College of Engineering & Technology
Mumbai, India
chiragnarkar2507@gmail.com

Harsh Kore
Dept. of Information Technology
Vidyavardhini's College of Engineering & Technology
Mumbai, India
harshbkore@gmail.com

***Abstract* - Public shaming in online social networks and associated online public boards like Twitter, Facebook, and Instagram have been growing in recent years. These occasions are acknowledged to personally have a devastating effect on the victim's social, political, and monetary life. Notwithstanding its acknowledged sick effects, little has been finished with infamous online social media to treat this, regularly with the aid of using the excuse of giant extent and form of such feedback and, therefore, an unfeasible wide variety of human moderators required to acquire the project. In this research, we automate the project of public shaming detection in social media structures from the mindset of sufferers and discover ordinary aspects, namely, occasions and shamers. It is discovered that out of all of the collaborating customers who publish feedback in a completely specific shaming event, the bulk of them can disgrace the victim. Interestingly, it is also the shamers whose follower counts boom quicker than that of the non-shamers in several social media. Finally, primarily based totally on categorization and type of shaming tweets/feedback, a web utility has been designed and deployed for on-the-fly muting/blockading of shamers attacking a victim.**

Keywords— Public Shaming, Social-Media, Shamers Victims, Random Forst Classifier.

I. INTRODUCTION

Everywhere, every human being has gone through an insult or a shame be it personally or on a public platform. Shaming includes abusing, discriminating based on color, race, gender, sexual preference, religion, political opinions, etc. This rapid growth in the number of people connecting worldwide, especially on public platforms has led to growth in the number of hate speech, shaming, and cyberbullying. A few examples are "Women are nothing more than a scumbag", "Shit Person you don't deserve to live". This increase in the number is nothing less than an alarm about how humankind has become insensitive. As a result, there is a need in detecting online shaming and automate the process to mitigate it. This field has been progressed and taken into consideration due to the grievousness of this shaming problem. This will help to know the aspect of shaming, the zone of remark, various visuals, periodic aspects of posting, targeted victim, a community of culprit, etc. This paper presents to you the contribution done for detecting the shaming done on social media platforms and taking actions against the user and making this whole process automated.

Every individual is targeted online whether they have done something wrong or not. Even its normal human tendencies to have differences of opinion on a topic but are cursed, abused on the difference of opinion. One can hide their identity on various public platforms and can harass them to take any drastic step. [1] One can also influence an opinion that may lead to making wrong decisions sometimes. Moreover, for a limited number of comments, you can personally delete/ block the particular comment. But comment sections for public figures have a multitudinous number of comments and dealing with it manually is something that we can say is a tedious job. The paper's purpose is to learn how to make such a tedious job easy by using various Machine Learning algorithms by constructing a public shaming analyzer machine learning tool that gives you the list of comments which will classify is it abusive/ shaming or not.

The first extraction of the comments from a particular account will be done after going through the comment section of all the posts of a particular account, by using a machine trained by one of the algorithms of machine learning it will be classified as shaming or not.

II. REVIEW OF LITERATURE

The paper researched by Dr. Andrew and Bridianee mentions the given points, on social networking sites, people tend to cross their boundaries having the misconception that no actions can be taken against them. They insult, abuse, and shame someone without thinking about the consequences. But according to the survey, actions were taken against them on a large scale which resulted in a loss of jobs or a losing position from a reputed committee [3] To achieve this progress has been made using different ways. Firstly, talking about the dataset, many datasets are available on open-source platforms. Waseem and Hovy released a public data set of 16,000 tweets labeled in one of the three categories: racist, sexist, or none in their research paper Hateful Symbols or Hateful People. They achieved an F1 score of 0.73 using character n-grams with logistic regression.

Similarly, work done by P Badiatiya provides an F1 score of 0.93 using deep neural networks on the same dataset. [4] Manually annotations were done to bring them into the desired format. Word generalization, stemming and various other features are used to clean the dataset.

Adolescent's minds tend to learn vulgar and explicit content quickly which may have an impact on their minds, to counter this the authors of "Detecting offensive language in

 ISSN 0973-7529; ISBN 978-93-80544-45-8

social media to protect adolescent online safety" suggest using different methods to mitigate various shaming related issues on social sites, and further predict a user's potentiality to send out offensive contents which are part of our research as well.

Different approaches have been seen. Machine learning classifiers have been commonly used to classify into different categories like sarcasm, abuse, comparison, and many others. Other methods like deep learning, natural language processing, and neural networks have been implemented to train and test datasets.

III. Method

In this section, we present our procedure for making a public shaming analyzer with the highest accuracy possible. The procedure is divided into smaller steps to attain maximum accuracy, success, and is free of bugs. The main objective of this project is to analyze social media comments. The entire project is divided into three major steps:

- Scraping of the comments
- Pre-processing of data acquired
- Training and Testing of the Machine Learning Model.

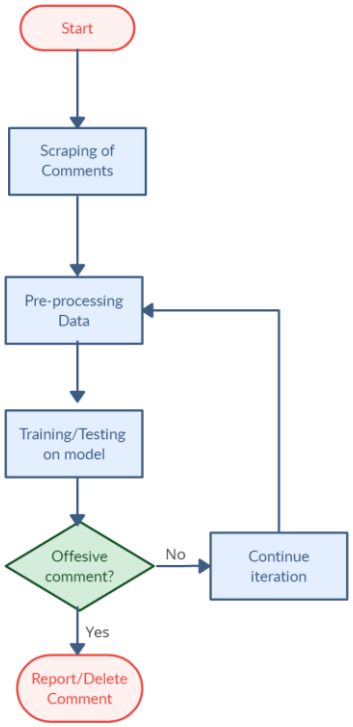

Fig. 1. Procedure Flow

A. Scraping of the comments

The primary step of our project is to get real-time comments on a particular account. It was achieved by selenium for scraping real-time comments. Selenium helps to automate the testing of web applications. Additionally, we need a chrome driver to get access and control of the chrome web browser as the algorithm visits each post's URL, and then the comments are extracted.

B. Training and testing of the Machine Learning model

The next and most important step of the project is to train and test the model. For Training our model we have used a random forest algorithm [4]. For training, there were many dataset on open source, combining it and manually annotations were done to train from every aspect. Dataset was split in an 80:20 ratio where 80 were used in training and 20 was used in testing.

IV. Solution Approach

The project is divided into three steps. This section gives information about the detailed work done.

A. Scraping of the comments

To get real-time comments this is a list of things that were used:

Selenium - For Automating our web app on browsers selenium is used. Selenium has different functions. For the research, we have used a selenium web driver.

Selenium- web driver - In simple terms, the selenium web driver automates and controls your browser. It helps you to connect your code to the site given in your command prompt and complete further actions. For automation of the browser, selenium, and selenium- web driver was used but for actually extracting we used our approach which extracts the comments from a particular profile using the page source of the currently open post in the browser. First, we studied the page source of a particular social media platform. Every account must have a post and every post must have comments. So after studying we observed that every single post's link is unique but written in the same format for example "instagram.com/p/CVbCJploIcc/", "https://www.instagram.com/p/CValA8hoqOU/". As it is seen two posts of the same account have links that have unique values (CVbCJploIcc, CValA8hoqOU) whereas the common thing is it starts with "/p/" so considering that we started splitting source code and took the link of every post. Now every comment has a unique ID so again splitting the source code we extracted the comments one by one.

B. Pre-processing, Training, and Testing of the model

1) Pre-processing

After getting the raw comments from a social media profile the next step is pre-processing. [4] This step is important as the dataset on which the model is to be trained contains a lot of noise and elements which either don't play role in determining the nature of the comment or are just merely an outlier, for example: "How is the life going in Seattle "here the stop words ('is', 'the', 'in') don't hold weight hence can be ignored. This removal of "Stop Words" is further explained with other pre-processing techniques.

Stop-words

These are the words that are the most commonly used in a language. In English the stop words can be "the", "us", "our", "in" etc. [4] All these types of words need to be removed as we are working on text classification, these words don't give us any information about the text and hence can't be given to a model for training and restricting the unwanted data from our corpus.

Stemming

This is the process of producing the variants of a root word or reducing a word to its root word. Considering an example, the words “Likes”, “Liked”, “Likely” and “Liking” can be reduced to their root words which are “Like”. But this cannot be applied in every case as it is prone to being erroneous hence lemmatization is also employed.

Lemmatization

This approach is similar to stemming i.e., reducing the words to their root words, but lemmatization is much more widely applicable as while converting the words to their roots the context of the original words is also considered and taken care of. Let’s say we have the word “Caring” if stemming is used the root word comes out to be “Car” but if the lemmatization approach is used the output is changed to “care” and it is evident from here that the context of the original words is preserved.

Tokenization

It is a process of splitting a sentence or paragraph into small units called tokens. This is an important step if we want to get the meaning of the sentence given, as the words present in the sentence gives us the meaning of the sentence rather than considering a whole sentence. For example, “Technology is Good” can be tokenized into [“Technology”, “is”, “Good”]. This helps us to determine the number of words in the sentence, it can also help us get information about the frequency of a particular word in the sentence.

Numeric and Special character removal

As we are doing text analysis, numeric values and special characters don’t play a major role in determining the meaning of the sentence hence has to be removed from the raw message.

Normalizing the slang

While dealing with the social media comments it is very obvious to have slang language as a part of a raw comment, but the model can’t be trained on the slang words hence it becomes important to convert slang into [6] original words so that they can provide us with the information which can be used for deciding future. For example, “I luv u” has to be converted to “I love you” so that it makes sense. To achieve this, we need a huge dictionary of slang words with their original words. Here is a small version of it slang dict=’luv’:’love’,’wud’:’would’,’lyk’:’like’,’wateva’:’whatever’,’ttyl’:’talk to you later’, ’kul’:’cool’, ’fyn’:’fine’, ’omg’:’oh mygod!’, ’fam’:’family’, ’bruh’:’brother’, ’cud’:’could’, ’fud’:’food’

Separating Hashtags

As we have comments as our inputs it is not surprising to have hashtags, [7] we have to store hashtags separately as they generally hold weight in determining the context of a sentence as positive or negative.

C. Training and Testing

For training purposes, there are many classification algorithms among which we have used random forest algorithms as they gave high accuracy when it came to text analysis.

Random Forest Classifier

It is a machine learning classifier that is mostly known for regression and classification-related problems. It gives results to complex problems by combining many classifiers randomly [9]. Decision trees are the bottom-up approach of random forest algorithms. The components of decision trees are root nodes, decision nodes, and leaf nodes. The dataset is categorized into different branches by the decision trees until it reaches the leaf node. There are several decision trees in a random forest.

Fig. 2. Negative Words

Fig. 3. Normal Words

Random forest is trained through bagging. The outcome of a random forest is based on the highest number of votes given. Different decision trees give different predictions and the prediction with the highest number is selected and given as output. In short Random Forest algorithm is known for giving a single result by taking the output of multiple decision trees. Important parameters that affect the output are node size the number of trees, and the number of features sampled. One of the benefits of the random forest algorithm is it reduces the risk of overfitting.

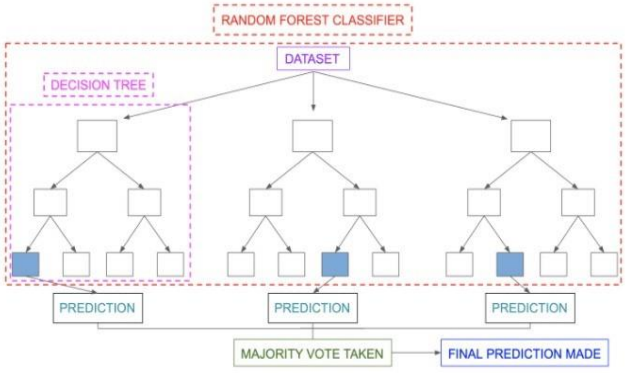

Fig. 4. Random Forest Architecture

In finance and healthcare, random forest is trusted because of the accuracy given. On testing this algorithm, we got an accuracy of 0.94(94percent) with stemming and lemmatization and 0.83(83 percent) without stemming and lemmatization below is the report attached.

V. Result and Decision

Result of implementing various steps, as it is impossible to cover every emotion of a comment. After implementation, these are results given on various steps of input. Following are some of the implementation results (Figure 5 to 11).

```
...  Random Forest Classifier Report
     Predicted     0     1
     Actual
     0          3221   166
     1           166  2122
                   precision    recall  f1-score   support

                0       0.95      0.95      0.95      3387
                1       0.93      0.93      0.93      2288

         accuracy                           0.94      5675
        macro avg       0.94      0.94      0.94      5675
     weighted avg       0.94      0.94      0.94      5675
```

Fig. 5. Random Forest report (with stemming and lemmatization)

```
Random Forest Classifier Report (Without Stemming and Lammetization)
Predicted     0     1
Actual
0          3215   172
1           168  2120
              precision    recall  f1-score   support

           0       0.80      0.85      0.84      3387
           1       0.82      0.82      0.81      2288

    accuracy                           0.83      5675
   macro avg       0.84      0.84      0.86      5675
weighted avg       0.84      0.84      0.86      5675
```

Fig. 6. Random Forest report (without stemming and lemmatization)

Fig. 7. Actual Comments (Instagram)

Fig. 8. Comment Extracted (Instagram)

Fig. 9. Actual Comments (Facebook)

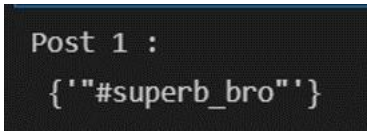

Fig. 10. Comment Extracted (Facebook)

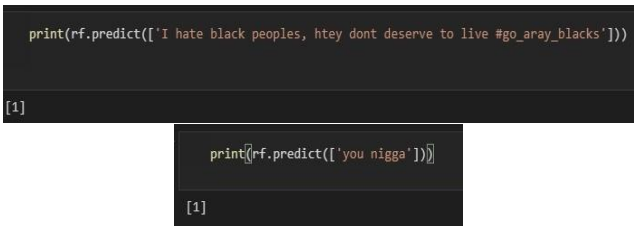

Fig. 11. Real-time comment classified as hate speech

VI. Future Work

For future work, the current project only analyze the comments, but in the future, we can predict the personality of a user based on their comment section, the content posted, and their search section and this can be achieved using the deep learning method [10] also the various method can be implemented for improving accuracy in random forest method [11].

VII. Conclusion

In recent days there are several studies have been done on text summarization but using text summarization for cyberbullying needs more attention as the growth of cyberbullying and hate speech on social media is a common case now. Analyzing the comments on a particular social media post can be helpful in a variety of cases as it can be used on a personal level for your social platform integrity or it can be used by a business, in general, to monitor their public image as it becomes important as well as difficult when it is a big business with a lot of public interaction. In this research, we have implemented a Random Forest algorithm for text analyzing the result we got without Stemming and Lemmatization is 0.83 (I.e., 83 percentage) which is good but the accuracy can be drastically increased using stemming and lemmatization which is 0.94 (i.e., 94 percentage). Hence it can be concluded that Random Forest can be a great approach in text summarization and can be used in the future for further work related to the same, it is likely that in near future, automatic cyberbullying detection can be excellent and useful as compared to manual checks.

ACKNOWLEDGEMENT

We would like to express heartfelt gratitude to my internal guide, Prof. Swati Verma, for rendering all possible help and support for the implementation of the idea successfully. We would also like to thank our parents and friends that helped to complete the paper in a limited time frame. I am equally grateful to our faculty of management for their support. This paper includes collective efforts and dedication from our group members. The success and outcome required a lot of guidance from many people and we are very fortunate to have got all this help

REFERENCES

[1] Bridianee O'DEA, Dr. Andrew CAMPBELL, "Online Social Networking and the Experience of Cyber-Bullying" Annual Review of Cybertherapy and Telemedicine 2012.

[2] Anna Schmidt, Michael Wiegand, "A Survey on Hate Speech Detection using Natural Language Processing" Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, Valencia, Spain, April 3-7, 2017.

[3] Rajesh Basak, Shamik Sural, Niloy Ganguly, Soumya K. Ghosh "Online Public Shaming on Twitter: Detection, Analysis, and Mitigation", IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS, VOL. 6, NO. 2, APRIL 2019

[4] Zeerak Waseem, Dirk Hovy "Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter," Proceedings of NAACL-HLT 2016, pages 88–93.

[5] Thomas Davidson, Dana Warmsley, Michael Macy, Ingmar Weber, "Automated Hate Speech Detection and the Problem of Offensive Language" arXiv:1703.04009v1 [cs.CL] 11 Mar 2017.

[6] Ying Chen, Sencun Zhu, Yilu Zhou, Heng Xu, ' Detecting offensive language in social media to protect adolescent online safety'.

[7] Armand Joulin, Edouard Grave, Piotr Bojanowski, Tomas Mikolov, 'Bag of Tricks for Efficient Text Classification', Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics.

[8] Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, Yi Chang', "Abusive Language Detection in Online User Content", International World Wide Web Conference Committee, April 11–15, 2016, Montréal, Québec, Canada.

[9] Gerard Biau, "Analysis of a Random Forests Model", Journal of Machine Learning Research 13 (2012) 1063-1095.

[10] Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, Vasudeva Varma. "Deep Learning for Hate Speech Detection in Tweets." 2017 International World Wide Web Conference Committee.

[11] Kristiawan Nugroho, Edy Noersasongko, Ahmad Zainul Fanani, Ruri Suko Basuki 'Improving Random Forest Method to Detect Hatespeech and Offensive Word' , 2019 International Conference on Information and Communications Technology (ICOIACT).

[12] https://en.wikipedia.org/wiki/Random_forest#/media/File:Random_forest_diagram_complete.png.

Swati BE Group 25

ORIGINALITY REPORT

15%	14%	8%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

MATCH ALL SOURCES (ONLY SELECTED SOURCE PRINTED)

7%

★ www.coursehero.com

Internet Source

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		