SSM: Cyberbullying detection and prevention on social network

by

John Hani, Mohamed Nashaat, Mostafa Ahmed, Zeyad Emad

A dissertation submitted in partial fulfillment of the
requirements for the degree of
Bachelor of computer science

in

Department of Computer Science

in the

Faculty of Computer Science
of the
Misr InternationalUniversity, EGYPT

Thesis advisor:
Dr. Eslam Amer, Dr. Ammar Mohamed, Eng. Menna Gamil

(June 2019)

Abstract

It's a good thing to have ways of communication between you and friends, but such communication could backfire on anyone. With the rapid increase of social media usage covering the cyber space, people will start to question if it's ever enough. More over, people also question the type of content they see, and the kind of messages they're receiving. Such deliberate texts are meant for a purpose which is to harm others without the need of physical contact. Cyberbullying has become a major concern nowadays and it has become much easier thanks to the fertile land of social media. Anyone could be anything, the good and the bad.

Although, it's not easy to detect cyberbullying on social media due to neglecting the polarity of the chat history which results in high rate of false positive due to ignoring the nature and background of these texts. In our paper, we propose an approach specific for detecting cyberbullying while chatting. Compared with other methods of cyberbullying detection, we have the best performance in detection of such cases with the lowest rate of false positive.

Acknowledgments

We are heartily thankful to Professors Ammar Mohamed and Eslam Amer in the faculty of Computer Science at Misr International University for their efforts as our mentors and being our 5th member of the team their experience and work influence have really affected our performance as a whole pushing us forward through making constructive comments and decisive actions that helped push the project to what it is now, their office was always open to us, answering our every question, and throughout the entire year every phase delivery has been productive on both ends.

We would also like to thank Dr. Ayman Ezzat in the faculty of Computer Science at Misr International University for all the effort he has put into each and every graduation project at Misr International University and for his participation and acknowledgement with feedback and constructive criticism and for his frequent advisory.

A sincere thanks goes to Dr. Ashraf Abdelraouf in the faculty of Computer Science at Misr International University for his dedication and vast knowledge throughout each and every discussion from the proposal at the beginning to the final presentation at the very end.

We will always be in gratitude of Dr. Ayman Nabil for helping the team in numerous ways such as gathering the tools necessary for applying in competitions and aiding the team financially for publishing our paper in prestigious journals.

And Dr. Ayman Bahaa Dean of the faculty of Computer Science at Misr International University for providing us with years upon years of experience and valuable knowledge that only adds more appraisal in our work and for his on going work in further improving the faculty's worth and essence.

Last but not least Dr. Reeham Abd El Razek the Psychologist for further helping us understand the the magnitude of the problem we're addressing by adding an expertise opinion and thoughts of our work and help us in understanding what solutions we're offering for the current issues at hand.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction

1.1.1 Background

Social media activity has increased in the Middle East over the last decade. According to "social media in the Middle East in 2017" the number of the active social media users are 93 million people a day. As we all know social media platforms are a good place for communication, sharing information and maintaining the old relationships. On the other hand, it affects the society in a negative way especially the teenagers. One of the biggest impacts is cyberbullying. Cyberbullying can lead to many psychological, physical and mental effects like feeling lonely, depression, anxiety, and the dangerous thing is that bullying can lead to suicide. So due to the prevalence of cyberbullying according to bullyingstatistics.org over half of the youth have been cyberbullied and equal to this number have been involved in cyberbullying. Our aim is to detect cyberbullying in the existence of sarcasm using machine learning classifiers and deep learning classifiers.

1.1.2 Motivation

In the last couple of years, the usage of the internet and social media has increased drastically and this usage will continue to grow over time, with this increase, the amount of cyberbullying will be huge in the near future. There are many non-profit organizations that call to stop cyberbullying like UNICEF is now making a great campaign to eliminate

bullying in Egypt. As we mentioned before, the community is suffering from cyberbullying and some action has to be done to prevent it.

1.1.3 Problem Definitions

Data on internet nowadays is too huge to be monitored manually by humans to detect cyberbullying. In previous cyberbullying detection papers there has been a problem in detecting false positive cyberbullying cases. The accuracy in these papers is not high enough and could be improved also no functional application was made to ensure portability.

1.2 Project Description

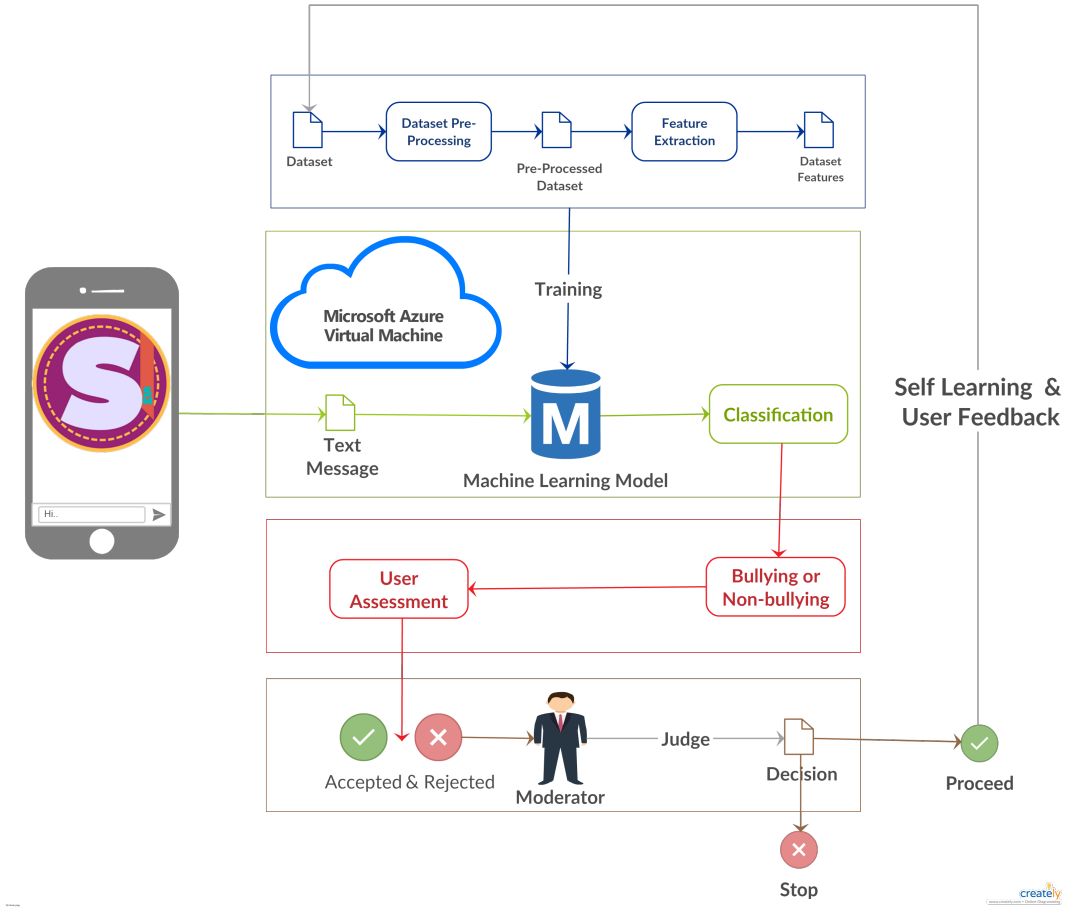
1.2.1 Objective

Our goal is to build an application thatâĂŹs able to detect and prevent cyberbullying using sentiment and TFIDF and machine learning classification. First, we want to make the system able to understand the message in chat to detect emotions in the posts and decide whether there is cyberbullying or not. Finally, when a cyberbullying is detected the user can rate the the detection of the message then the administrator of the social media platforms to take the appropriate action. The system will only cost the cloud server hosting and a laptop to develop.

1.2.2 Scope

The system will cover in its scope: 1. Sentiment analysis of text combined with TFIDF. 2. The system will work on word level analysis and also phrase level analysis. 3. The system will use machine learning classifiers to increase the accuracy

1.2.3 Project Overview



1.3 Project Management and Deliverable

1.3.1 Tasks and time plan

Task Name	Start Time	Finish
Idea Discussion	1/8/2018	1/8/2018
Idea Research	1/8/2018	13/9/2018
Proposal Writing	13/9/2018	16/9/2018
Implementing Prototype	16/9/2018	17/9/2018
Delivering Rehearsal	18/9/2018	18/9/2018
Delivering Proposal	18/9/2018	26/9/2018
Doing Survey	10/10/2018	20/10/2018
Implementing Second Prototype	20/10/2018	25/10/2018
Writing SRS	25/10/2018	30/10/2018
Implementing	30/10/2018	25/11/2018
Preparing For External Examiner	25/11/2018	3/12/2018
Implementing	3/12/2018	18/1/2019
Writing SDD	18/1/2019	1/2/2019
Implementing	1/2/2019	1/4/2019
Preparing For Implementation Evaluation	1/4/2019	25/4/2019
Writing 8 Pages Paper	25/4/2019	28/4/2019
Finalizing Implementation	28/4/2019	7/5/2019
Writing Final Thesis	10/5/2019	25/5/2019
Presenting Final Thesis	25/6/2019	25/6/2019

1.3.2 Budget and Resources Costs

- Cloud server 28$/month.
- Laptop 8000 EGP.

Chapter 2

Literature Work

2.1 Similar System Information

1. Sentiment Informed Cyberbullying Detection in Social Media [9]:

   (a) In this paper the researchers were motivated by psychological and sociological findings, wanted to investigate the relationship between sentiment information and Cyberbullying behaviors.

   (b) The main problem is addressed used sentiment analysis to detect cyberbullying and dealt with 2 problems: short, noisy and unstructured content information and the obfuscation of the obnoxious words by the users.

   (c) Researchers proposed a principle learning framework called (SICD) and they study whether sentiment information is particularly correlated with cyberbullying behaviors and how to deal with short and unstructured content.

   (d) Researchers conducted extensive experiments on two real-world data-sets. The experimental results show the effectiveness of the pro-posed model as well as the impact of sentiment information.

   (e) This Paper is going to help us in the sentiment analysis section as they held out many experiments that investigate the effectiveness of the sentiment analysis on cyberbullying.

2. Automatic Detection of Cyberbullying on Social Networks based on Bullying Features[26]:

(a) They made this program because the increasing of social media data which increase the cyberbullying that give bad impacts on children and teenagers such as depression and suicidal thoughts.

(b) The main problem of BOW is that every word is independent from the other and that fails to see the sentence as a whole.

(c) They made a framework that detect the cyberbullying, based on word embedding, they made a list of insulting words then they assign weights to them. After this they concatenate latent semantic feature with bag of words then they classified them with SVM.

(d)

Measures	BoW	sBoW	LSA	LDA	EBoW
Precision	75.6	75.7	75.9	74.0	**76.8**
Recall	77.8	78.3	78.2	76.5	**79.4**
F1 Score	76.6	76.9	77.0	74.9	**78.0**

(e) It is important to us because they concatenate bag of words with latent semantic feature.

3. Cybercrime detection in online communications: The experimental Case of cyberbullying detection in the Twitter network [**?**]:

(a) The bad effects of social media like cyberbullying that make the cyberbullied person suffering from many things such as suicidal thoughts and depression.

(b) They donâĂŹt have word embedding or sentiment analysis they rely their work on classification.

(c) Their model takes network, tweet content, activity and user features from tweets then they train random forest with SMOTE classifiers to classify cyberbullying and non-cyberbullying.

(d) Results: under the receiver operating characteristic (ROC) curve (AUC) of 0.943 fmeasure of 0.936 using random forest with SMOTE.

(e) It is important to us because they use hybrid classifiers which one of them is random forest and we plan to use these methods.

4. Unsupervised Cyberbullying Detection in Social Networks [**?**]:

(a) While cyberbullying is a well-studied problem from a social point of view, only recently it has attracted the attention of computer scientists, especially towards automatic detection tasks. For this reason, only relatively few articles on the subject and very few datasets are available.

(b) We proposed to adopt an unsupervised approach to detect cyberbullying traces over social networks.

(c)

TABLE I
RESULTS OBTAINED ON FORMSPRING.ME DATASET

Precision	Accuracy	Recall	F1	Method
0.72	0.73	0.69	0.71	GHSOM
0.60	-	0.40	-	C4.5
-	-	0.67	-	SVM

TABLE II
AVERAGE RESULTS OBTAINED ON YOUTUBE DATASET.

Precision	Accuracy	Recall	F1	Method
0.60	0.69	0.94	0.74	GHSOM

TABLE III
AVERAGE RESULTS OBTAINED ON TWITTER DATASET.

Precision	Accuracy	Recall	F1	Method
0.81	0.72	0.26	0.4	GHSOM
-	0.67	-	-	Naive Bayes

(d) We now know multiple sources that we can setup as our data sets (YouTube, twitter, FormSpring)

5. Experts and Machines against Bullies: A Hybrid Approach to Detect Cyberbullies [7]:

(a) Most of the technical studies have focused on the detection of cyberbullying through identifying harassing comments rather than preventing the incidents by detecting the bullies.

(b) Proposed methods: we introduce the three types of models used for calculating and assigning the bully score to the social network users: a multi-criteria evaluation system, a set of machine learning models and two hybrid models that combine the two.

(c) Machine Learning Approaches: We used three well-known machine learning methods, which use pre-labelled training data for automatic learning: a Naive Bayes

classifier, a classifier based on decision trees and Support Vector Machines (SVM) with a linear kernel

(d) Results: The discrimination capacity of the MCES was 0.72.

6. Cyberbullying System Detection and Analysis [**?**]:

(a) Cyber-bullying has recently been reported as one that causes tremendous damage to society and economy.

(b) The system relies on the detection of three basic natural language components corresponding to Insults, Swears and Second Person pronoun.

(c) Proposed Methods: the whole is greater than the sum of its parts. A combination of moderate accurate features coming from heterogeneous data modalities can outperform methods that employ a single modality.

(d)

Feature	Acc.	Prec.	Reca.	F1-mea	F2-mea
tf-Idf	97,3%	31,2%	68,4%	42,85%	55.23%
LIWC	76,4%	28,4%	57,1%	32,56%	41,97%
Depen	67,5%	27,3%	60,6%	37,64%	48,72%
tf-Idf+LIWC	97,8%	42,4%	75,1%	54,20%	65,01%
LIWC + Depen	82,1%	38,4%	69,5%	49,47%	59,81%
tf-Idf+Depen	97,9%	58,9%	78,4%	67,26%	73,53%
All features	**99,4%**	**69,0%**	**84,9%**	**76,13%**	**81,15%**

(e) This work opens up new direction for future research through using advanced parser, dimension reduction and taking into account the userâĂŹs profile in order to strengthen the detection capabilities.

7. Common Sense Reasoning for Detection, Prevention, and Mitigation of Cyberbullying [**?**]:

(a) Cyberbullying or harassment on social networks is as much a threat to the viability of online social networks for youth today as spam once was to email in the early days of the internet.

(b) Proposed models: To detect explicit bullying language pertaining to (1) sexuality, (2) race and culture and (3) intelligence. Binary classifiers outperform their multiclass counterparts: JRip and Support Vector Machines were the best performing in terms of accuracy and kappa values.

(c)

	Naive Bayes			Rule-based JRip			Tree-based J48			SVM (poly-2 kernel)		
	Acc.	F1	kappa	Acc.	F1	kappa	Acc.	F1	Kappa	Acc.	F1	kappa
Sexuality	66%	0.67	0.657	**80%**	0.76	0.598	63%	0.57	0.573	66%	0.77	**0.79**
Race and Culture	66%	0.52	0.789	**68%**	0.55	0.789	63%	0.48	0.657	66%	0.63	**0.71**
Intelligence	72%	0.46	0.467	**70%**	0.51	0.512	70%	0.51	0.568	72%	0.58	**0.72**
Mixture	63%	0.57	0.445	63%	0.60	0.507	61%	0.58	0.456	66%	0.63	0.653

(d) Future work: They are currently embarking on the use of a family of latent variable models to model, understand and predict self-harm in adolescents, a phenomenon that is not very well understood in the field of abnormal psychology.

8. Improved Cyberbullying Detection Using Gender Information [**?**]:

(a) We used a supervised learning approach to detect cyberbullying. We constructed a Support Vector Machine classifier using WEKA.

(b) Four types of features: Profane words, second person pronouns, other personal pronouns, and the weight of the words in each sentence.

(c)

Table 1. The accuracy measures for basic and gender-based approaches for cyberbullying detection in a MySpace corpus

Feature used in classifier	Precision	Recall	F-measure
Baseline	0.31	0.15	0.20
Gender-specific	0.43*	0.16*	0.23*
Female-specific (34% corpus)	0.40	0.05	0.08
Male-specific (66% corpus)	0.44	0.21	0.28

(d) Future work: Considering contextual features of the text as well as the word level features. The ground truth annotation can be done through crowdsourcing, investigate other features which may differentiate the writing styles of the users such as age, profession, and educational level.

9. Machine Learning Approach for Detection of Cyber-Aggressive Comments by Peers on Social Media Network [4]:

(a) There is an enormous amount of information to manually flag offensive comments or posts. So an automatic classifier that is fast and effective is needed to solve this problem.

(b) Their problem is a binary classification problem where we are trying to classify comments as bullying and non- bullying.

(c) They proposed 2 new hypotheses for detecting cyberbullying and it has increased the precision by 4 %.

(d) The achieved 70 % precision using SVM classifier and 64 % precision using logistic regression.

10. A Pattern-Based Approach for Sarcasm Detection on Twitter [3]:

(a) Sarcasm is a sophisticated form of irony widely used in social networks and micro-blogging websites. It is usually used to convey implicit information within the message a person transmits.

(b) Recognizing sarcastic statements can be very useful to improve automatic sentiment analysis of data.

(c) They used NPL and SVM and for features extraction:sentiment-related features, punctuation-related features, syntactic AND semantic features and pattern-related features.

(d) Their proposed approach reaches an accuracy of 83.1% with a precision equal to 91.1%.

2.2 Comparison with Proposed Project

we evaluate and compare our classifiers on the proposed approach with the work of [4]. In this work, they used logistic regression and SVM for classification and used the same data. Moreover, we have calculated the average accuracy, recall, precision and F-score of our two classifiers. The summary of results is shown in table 5.1. To compare the work, it is found that our proposed NN model outperforms all other classifiers and is ranked as the best results in terms of average accuracy and F-Score achieving accuracy 91.76% and f-score 91.9%. In fig. 5.3 we are comparing between our best classifier with their best classifier in case of accuracy. Finally, here in fig. 5.2 we are comparing between our best classifier with their best classifier in case of F-Measure.

Table 2.1: COMPARISON WITH RELATED WORK

	Classifier	Avg. Accuracy	Avg. Recall	Avg. Precision	Avg. F-Score
Vikas S Chavan	Logistic regression	73.76	61.47%	64.4%	62.9%
	SVM	77.65%	58.29%	70.29%	63.7%
Current Results	Neural Network	**91.76%**	**91.7%**	**92.4%**	**91.9%**
	SVM	89.87%	90.1%	89.6%	89.8%

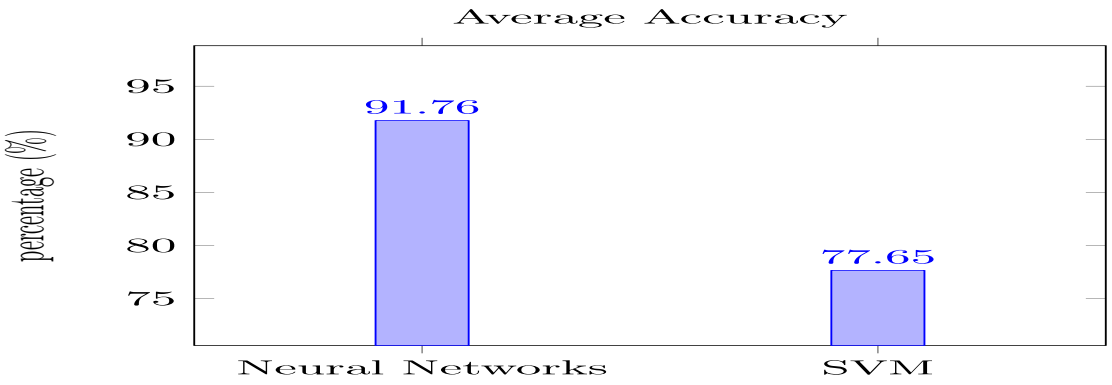


Figure 2.1: COMPARISON BETWEEN THE BEST CLASSIFIERS IN TERMS OF ACCURACY

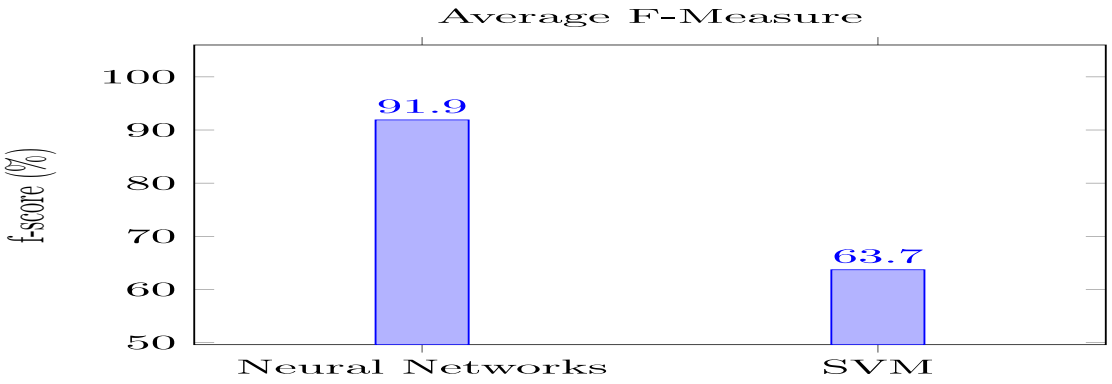


Figure 2.2: COMPARISON BETWEEN THE BEST CLASSIFIERS IN TERMS OF F-MEASURE

Chapter 3

System Requirements Specifications

3.1 Introduction

3.1.1 Purpose of this document

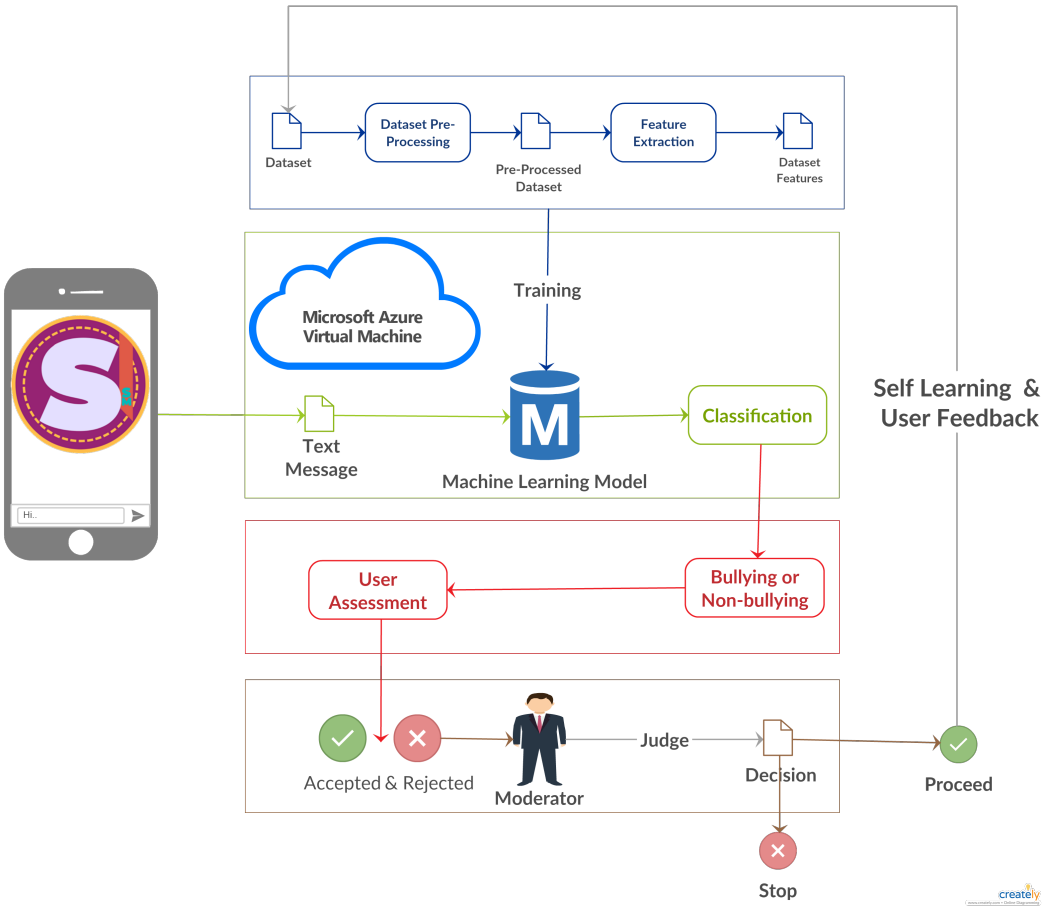
This document serves the purpose of the software requirement specification documentation and to provide a detailed overview of our software product, its parameters and goals and to describe the project's target audience and its user interface, hardware and software requirements.

3.1.2 Scope of this document

It defines how our client, team and audience see the product and its functionality. Our main concern here is that everyone gets the right idea as our functionality as presented to the audience is to develop a system which would be considered as a sub-system on a larger scale to detect and take further actions upon encountering cyberbullying. Our time line of delivery shall be June 2019, while it may cost 0 LE as it is presented for academic purposes, however that may change in the future while taking into consideration the market need. Nonetheless, it helps any designer and developer to assist in software delivery life cycle (SDLC) processes.

3.1.3 Overview

Depending on the social platform needs our developed program will be embedded within a larger system for cyberbullying detection and to make countermeasures regarding this



issue.

Message: the message will be received to our application from another person.

Smart phones: The application will be on smart phones that have internet connection and minimum RAM 512.

Cloud: The application will send the received message to the cloud .Then the message will go the classifier to detect that there is cyberbullying or not then the cloud will send the output to the application again to rate it.

Dataset: The Dateset size will increase with the help self-learning, as messages of wrong classification will be sent back once more to the dataset with the proper classification for retraining and refining after being revised by a supervisor.

3.1.3.1 User Management

3.1.3.1.1 This Module of the System tends to Manage the Use like Giving access to the Admins and Blocking Some users

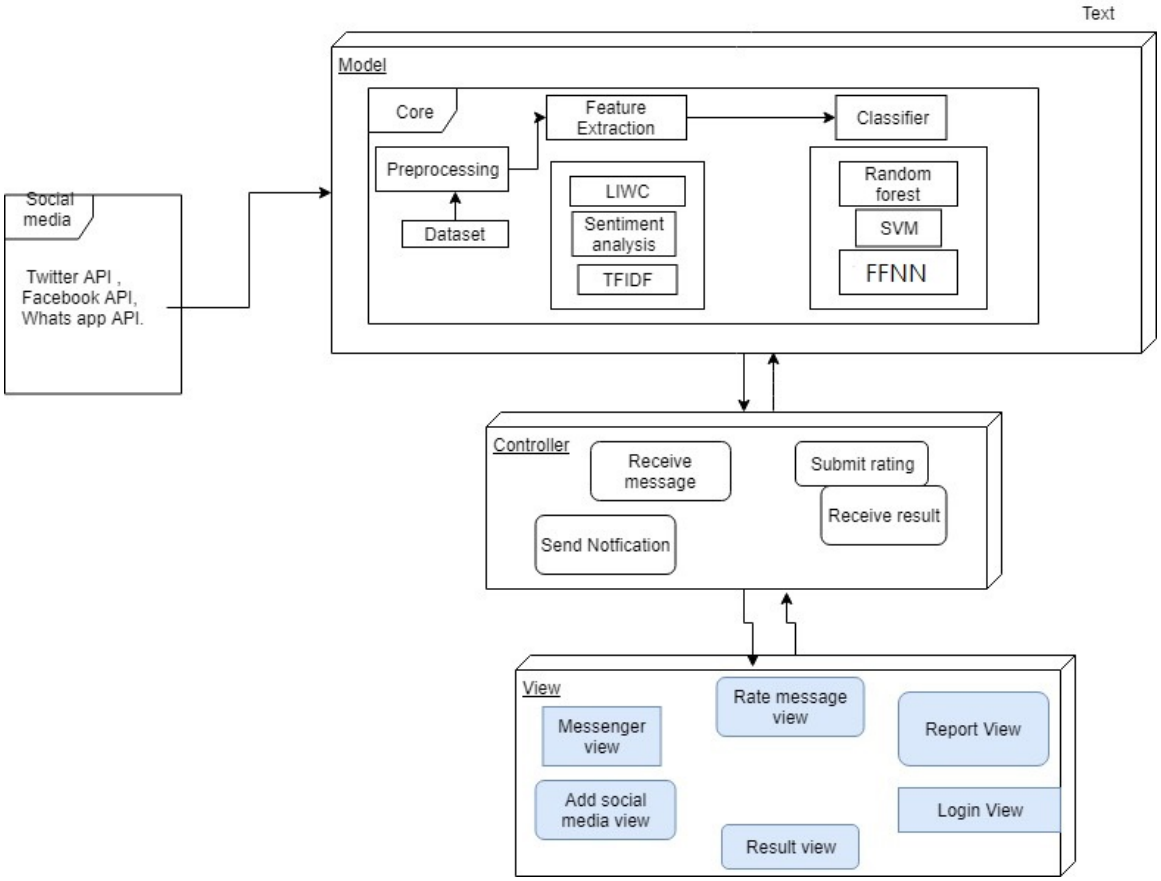


Figure 3.1: System Architecture

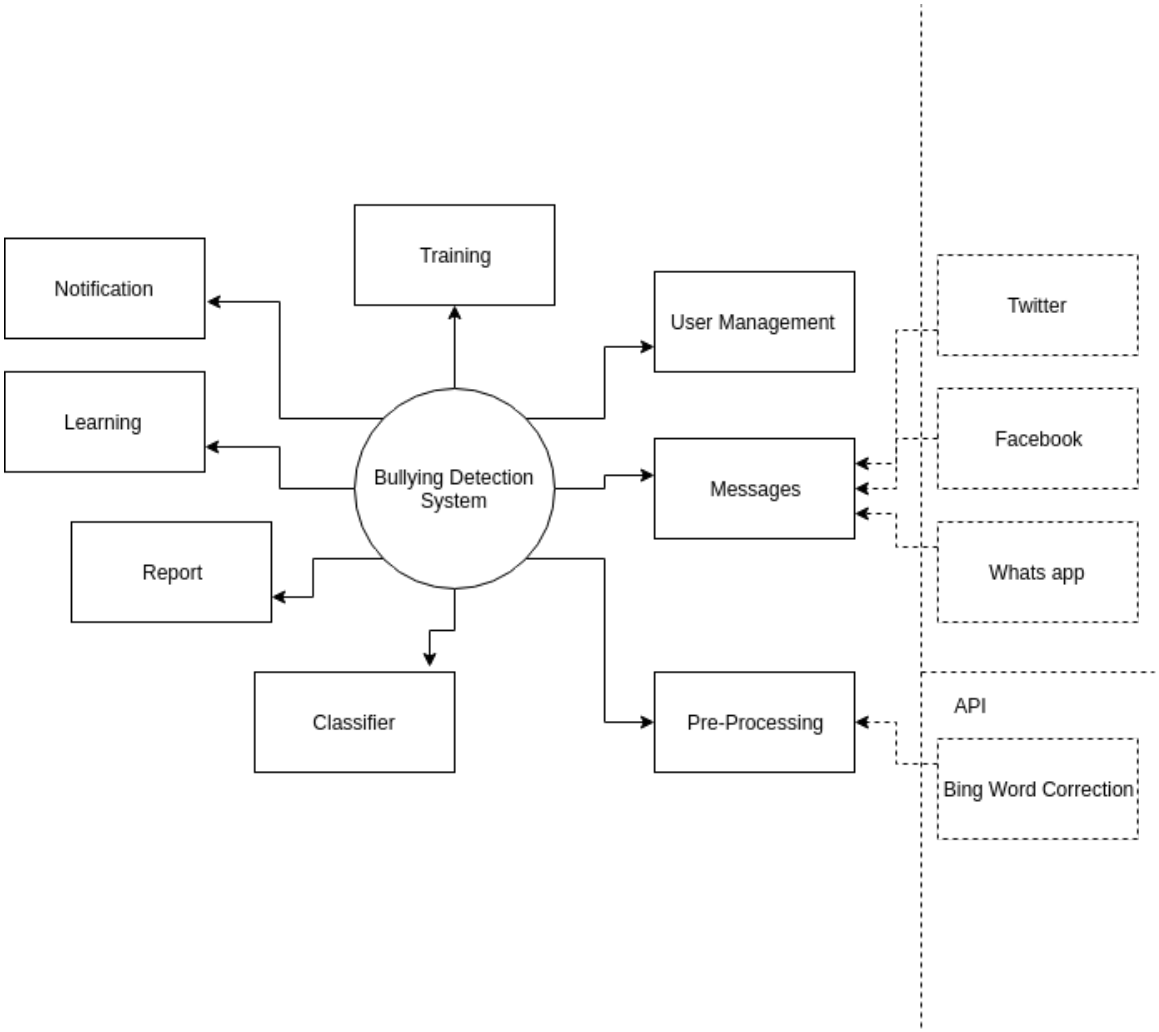


Figure 3.2: Context Diagram

3.1.3.2 Pre-Processing

3.1.3.2.1 The Pre-Processing Module is the first stage of processing the messages or posts before classification and feature extraction and the pre-procesing consists of Stemming,Limitization,Remove the Encoding Parts and the word correction where we use bing word correction API

3.1.3.3 Classifier

3.1.3.3.1 For the Classification we are going to have SVM classifier as our machine learning model, and neural network for our deep learning model.

3.1.3.4 Messages

3.1.3.4.1 The User application will receive messages

3.1.3.5 Report

3.1.3.5.1 Everyday the System is going to produce A detailed Report about the day events like (How many blocked bully messages, Average number of stars of the day, How many wrong classification made based on the user Rating

3.1.3.6 Learning

3.1.3.6.1 The Learning module is a self learning technique as messages of wrong classification will be sent back once more to the dataset with the proper classification for retraining and refining after being revised by a supervisor.

3.1.3.7 Notification

3.1.3.7.1 The system is going to notify the user of new messages and also notify that he/she received a bully message and it has been blocked

3.1.3.8 Training

3.1.3.8.1 The Training module is the module where the system learns either from a data set or from the approved wrong classified messages from the learn-

ing module and this module is adds more samples tp the classifier for better classification

3.1.4 Business Context

Moreover as to a business point of view more and more about cyberbullying will be dug through the vast extent of social media platforms as incidents of cyberbullying have doubled throughout the past 5 years.

1. Vision: Safe chatting for all people that are using social media.

2. Mission: Develop application that detect cyberbullying at the running time to help social media platforms and support the campaign that is happening nowadays that want to end all bullying if not then reduce it

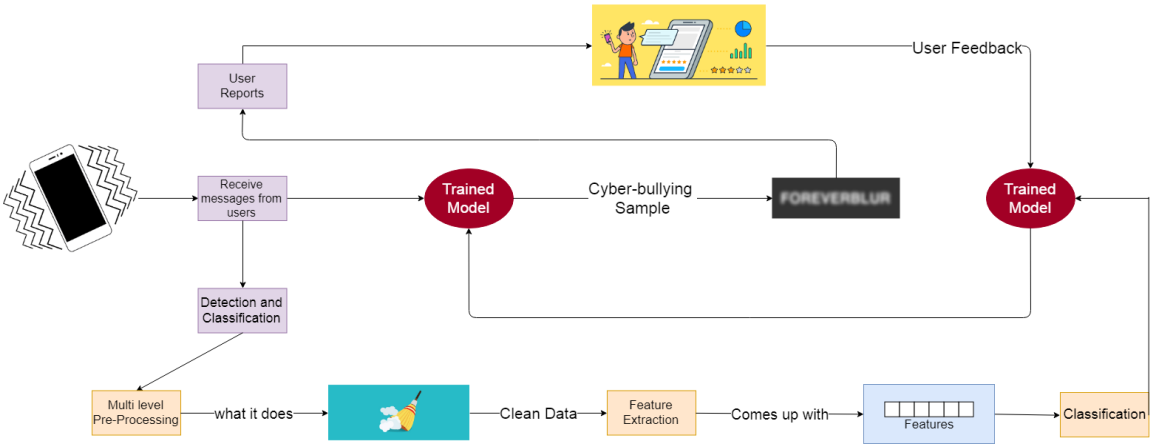


Figure 3.3: Business Model

3.2 General Description

3.2.1 Product Functions

Our bullying detection system consists of eight modules that are

1-Messages:It will receive this messages from Twitter or Facebook or Whats app to our application then it will send this message to the classifier.

2-Pre-processing: it will take the dataset or the message and it will cut them into sen-

tence then it will make ,Stemming, Lemmatization, Remove Encoding and Word correction using Bing word correction.

3-Training: There will be Training to the three classifiers that are SVM, Naive Bayes and Random forest.

4-Notification: The system will notify the client if there is cyberbullying in the message that comes to him and the system will notify the admin if there is error in the system.

5-Learning: the system will make the user rate the accuracy of its detection then this ratings will go the modulator to be sure that is cyberbullying then to train the classifier with the new messages.

6-Report:It will appear to the user of the system daily to show him the events that happened in these day.
7-User management: this module will have interaction with the user and it will deal with user Login, register and changing of user type.

8-Classifier this module will take message from message module and give result is it cyberbullying or not by voting between between three classifiers.

User Functions
1. sign in or 2. sign up as his role in the system as user
3. send message
4. Receive Message
5. void Rating(rate message)
As a customer he can update his own info. 6.Update
Admin will be able to see the errors
7.Message Errors

The developer can access the preprocessing and feature extraction and the processing

File manager functions

1. CSV
2. read Text

Pre-processing module

1. Read file.
2. Tokenization.
3. Stemming.
4. Lemmatization
5. Remove Encoding
6. Word correction

Feature extraction

1. TFIDF
2. LIWC
3. Sentiment features

Processing

(a) Fit()

(b) Prediction()

(c) Processing()

1. Training
2. SVM
3. Neural Network

The system will encrypt and decrypt the messages

User functions

The system will notify the user

Notification functions

1. Notify observer
2. Register observer
3. Remove observer
4. Update Notification

All User information will held and controlled by our database

Database Functions

1. add
2. delete
3. modify

3.2.2 Similar System Information

www.noswearing.com it is website that filters the sentences from curse or slang words and replace it with words have the same meaning of it .It is like the sites that change slang to normal language. They collect the bad words as the bag of words and they put it in a list so when the application find it .it will change it to another word with the same meaning.

Figure 3.4: Swear Word List and Curse Filter

3.2.3 User Characteristics

This project aims to target large number of people who use messenger applications. The users of our applications hope to not be bullied during their conversations on their messenger applications. Our application will detect the people who are making cyberbullying and report them.The user must be connected to the internet.

3.2.4 User Problem Statement

Data on the internet nowadays is too huge to be monitored manually by humans to detect cyberbullying and people use messenger applications multiple times per day so they

might be exposed to bullying messages. In previous cyberbullying detection papers there has been a problem in detecting false positive cyberbullyingâĂŹs cases. The accuracy in these papers is not high enough and could be improved. Sarcasm which is type of cyberbullying is not detected in these papers.

3.2.5 User Objectives

The application is designed for individual users who are using messenger applications, like WhatsApp and Facebook Messenger to detect bullying and sarcasm in messages. And the applications can report this messages.

3.2.6 General Constraints

Our system constraints is the ambiguity of words (Contextual Analysis) also the high rate of false positive instances. It is used mainly for smart phones, which are capable of having internet connection as well as having our unique Chatting application .

3.3 Functional Requirements

3.3.1 Send_Message

FRID	1
Description	This function sends Messages from user to the platform that the user signed in on and the function takes the message and id of user as a parameter
Action	Sends a message to the social media platform
Input	String and Integer
Output	Boolean
Precondition	the user must be logged in
Post-condition	None
Dependencies	
Priority	10 \10

Table 3.1: Send_Message

3.3.2 Receive_Message

FRID	2
Description	This function Receive Messages from the platform API
Action	Checks for any message sent to the user and then return the received message
Input	None
Output	Message object
Precondition	the user must be logged in
Post-condition	None
Dependencies	
Priority	10 \10

Table 3.2: Receive_Message

3.3.3 Preprocessing

FRID	3
Description	This function do some preprocessing on the recieved messages like: Stemming, Lemmatization, Tokenization, RemoveEncoding and WordCorrection
Action	Takes message as String and apply all of the preprocessing tools
Input	String
Output	one dimensional List
Precondition	There is text to process
Post-condition	The preprocessed text then goes to the feature extraction
Dependencies	
Priority	10 \10

Table 3.3: Preprocessing

3.3.4 Extract_Features

3.3.4.1 TFIDF

FRID	4
Description	This function extract features from the preprocessed text using TFIDF function
Action	Takes preprocessed text and extract features from it
Input	one dimensional List
Output	List of 2d numpy array
Precondition	There is preprocessed text to extract the feature from
Post-condition	Then goes to the classifier to classify the object
Dependencies	
Priority	10 \10

Table 3.4: TFIDF

3.3.4.2 LIWC

FRID	5
Description	This function extract features from the preprocessed text using LIWC function
Action	Takes preprocessed text and extract features from it
Input	[]List
Output	[]List
Precondition	There is preprocessed text to extract the feature from
Post-condition	Then goes to the classifier to classify the object
Dependencies	
Priority	10 \10

Table 3.5: LIWC

3.3.4.3 Sentiment_Analysis

FRID	6
Description	This function extract features from the preprocessed text using Sentiment Analysis
Action	Takes preprocessed text and extract features from it
Input	[]List
Output	[]List
Precondition	There is preprocessed text to extract the feature from
Post-condition	Then goes to the classifier to classify the object
Dependencies	
Priority	10 \10

Table 3.6: Sentiment_Analysis

3.3.5 Classification

3.3.5.1 Classify

FRID	7
Description	This function takes the extracted features of the text, train the classifier and predict the class of the sent message
Action	take the extracted features to train the classifier and classify the sent message
Input	two dimensional list for the text and one dimensional list for the classes
Output	None
Precondition	There is a sent message that needs to be classified
Post-condition	classify data
Dependencies	
Priority	10 \10

Table 3.7: Classification

3.3.6 SignUp

FRID	8
Description	This function creates account for users
Action	Takes Data of user and insert in database
Input	String firstName, String LastName,int age, String gender, String email
Output	Boolean
Precondition	That the user doesn't exists
Post-condition	Account created
Dependencies	
Priority	10 \10

Table 3.8: SignUp

3.3.7 Update_Classifier

FRID	9
Description	This function takes the extracted features of new data and updates the classifier with it
Action	Train the classifier with new data
Input	[][]list
Output	None
Precondition	There is new data to train the classifier with
Post-condition	Classify data
Dependencies	
Priority	10 \10

Table 3.9: Update_Classifier

3.3.8 Login

FRID	10
Description	This function verifies user's account
Action	Check for username and password in database
Input	String user-name & Sting password
Output	Boolean
Precondition	User has an account already
Post-condition	None
Dependencies	The application can't be accessed without being logged in
Priority	10 \10

Table 3.10: Login

3.3.9 Send_Notification

FRID	11
Description	This function sends notifications
Action	Sends notification from cloud to application
Input	String
Output	Boolean
Precondition	None
Post-condition	None
Dependencies	
Priority	10 \10

Table 3.11: Send_Notification

3.3.10 Encrypt

FRID	12
Description	This function encrypt messages
Action	Encrypt messages before being sent and encrypt user data before storing it in the database
Input	String
Output	String
Precondition	Data needed to be encrypted
Post-condition	Encrypted data
Dependencies	
Priority	9 \10

Table 3.12: Encrypt

3.3.11 Decrypt

FRID	13
Description	This function decrypt Data and messages
Action	Decrypt messages when the messages are received
Input	String
Output	String
Precondition	Data needed to be decrypted
Post-condition	Decrypted data
Dependencies	
Priority	9 \10

Table 3.13: Decrepit

3.3.12 Delete_Message

FRID	14
Description	This function deletes messages that the user selected
Action	Delete message from database and UI
Input	None
Output	Boolean
Precondition	The id exists in database
Post-condition	The message has been deleted
Dependencies	
Priority	8 \10

Table 3.14: Delete_Message

3.3.13 Copy_Message

FRID	15
Description	This function takes the selected messages and put them in clipboard for further operations
Action	Copies message to clipboard
Input	Message object
Output	Boolean
Precondition	THere is message selected
Post-condition	The message has been copied to the clipboard
Dependencies	
Priority	8 \10

Table 3.15: Copy_Message

3.3.14 Paste_Message

FRID	16
Description	This function paste messages
Action	print message from clipboard on screen
Input	Message object
Output	Boolean
Precondition	The id exists in clipboard
Post-condition	The message is pasted on screen
Dependencies	
Priority	8 \10

Table 3.16: Paste_Message

3.3.15 Logout

FRID	17
Description	This function logs the user out of the application
Action	Logout and redirect to login screen
Input	None
Output	Boolean
Precondition	The user must be logged in
Post-condition	Go to login screen
Dependencies	For the user to use the application he must login again
Priority	7 \10

Table 3.17: Logout

3.3.16 Show_Notification

FRID	18
Description	This function is for notification displaying
Action	Shows notification on screen
Input	String
Output	Boolean
Precondition	None
Post-condition	The notification is printed in screen
Dependencies	
Priority	7 \10

Table 3.18: Show_Notification

3.4 Interface Requirements

3.4.1 User Interfaces

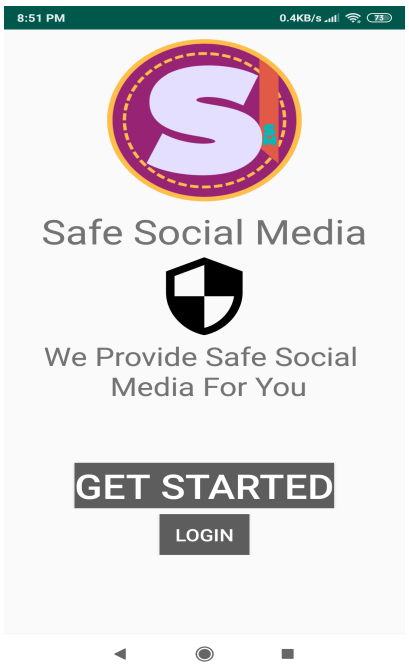


Figure 3.5: Rating

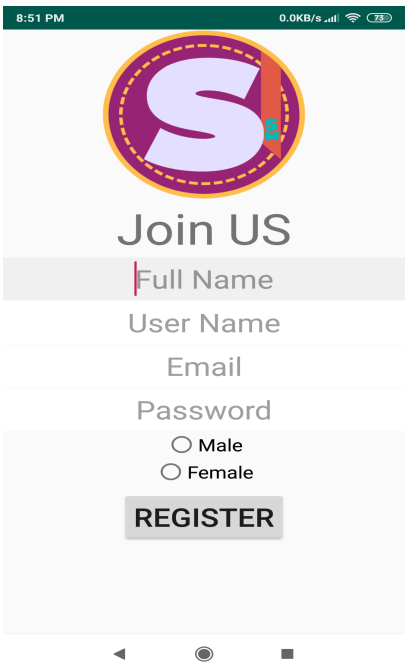


Figure 3.6: Register

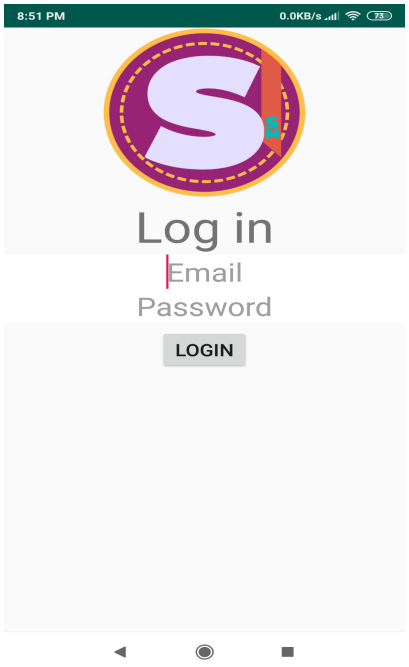


Figure 3.7: Log in

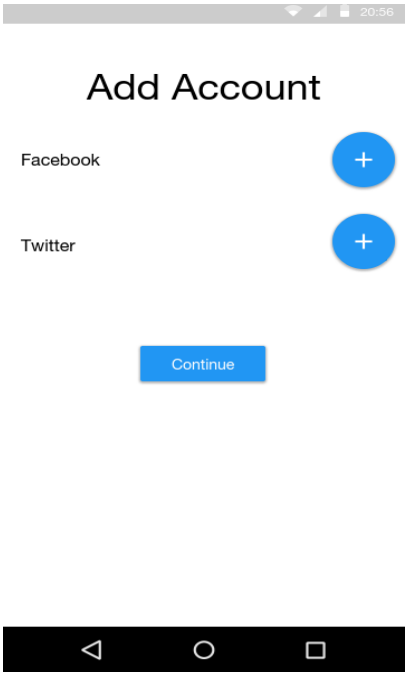


Figure 3.8: Add Social Media

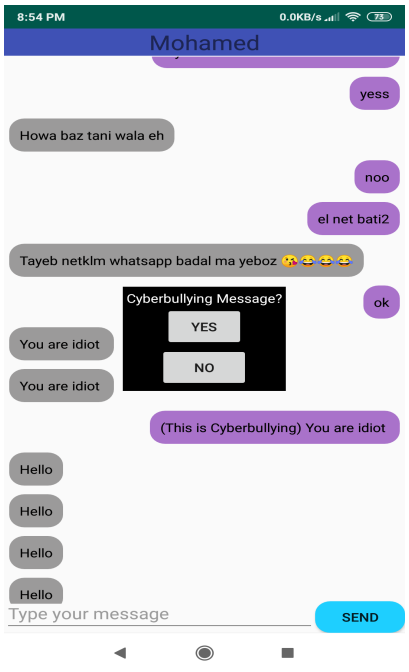


Figure 3.9: Rating

3.4.1.1 API

1. Bing word Correction API
2. Microsoft Azure VM API

3.5 Performance Requirements

The application will be fast as we know that the application will the send the messages to the cloud to detect if there is cyberbullying in the message or not. This process must never consume alot of time and we aim to make it in less than one second. Our application will use memory as the other freeware and cross-platform messaging service we estimate that it will use 100 MB as average usage.

3.6 Design Constraints

3.6.1 Standards Compliance

Our application will need an android device that is connected to the internet that we will install the application on it. And the application will be installed on the cloud.

3.6.2 Hardware Limitations

The phone must be at least 512 Ram also it must contain appropriate storage space to put the application. and it must be a smart phone.

3.6.3 others as appropriate

3.7 Other non-functional attributes

3.7.1 Security

The data between the user app and the host(Trained Model) Should be encrypted because it consists of Private Messages and also the login credentials for the user in our system should be highly encrypted.

3.7.2 Portability

The Software is going to be portable as it is going to be available for mobile environments and could make it support any platform as the core of the application is on a cloud and this application is just a client that sends the data for the core to process it.

3.7.3 Maintainability

Our Proposed Application should be maintainable as it learns its self from the previous wrong classifications by the help of the user feedback on miss-classified instances and also by the reporting of the user to the wrong classifications.

3.8 Preliminary Object-Oriented Domain Analysis

3.8.1 Inheritance Relationships

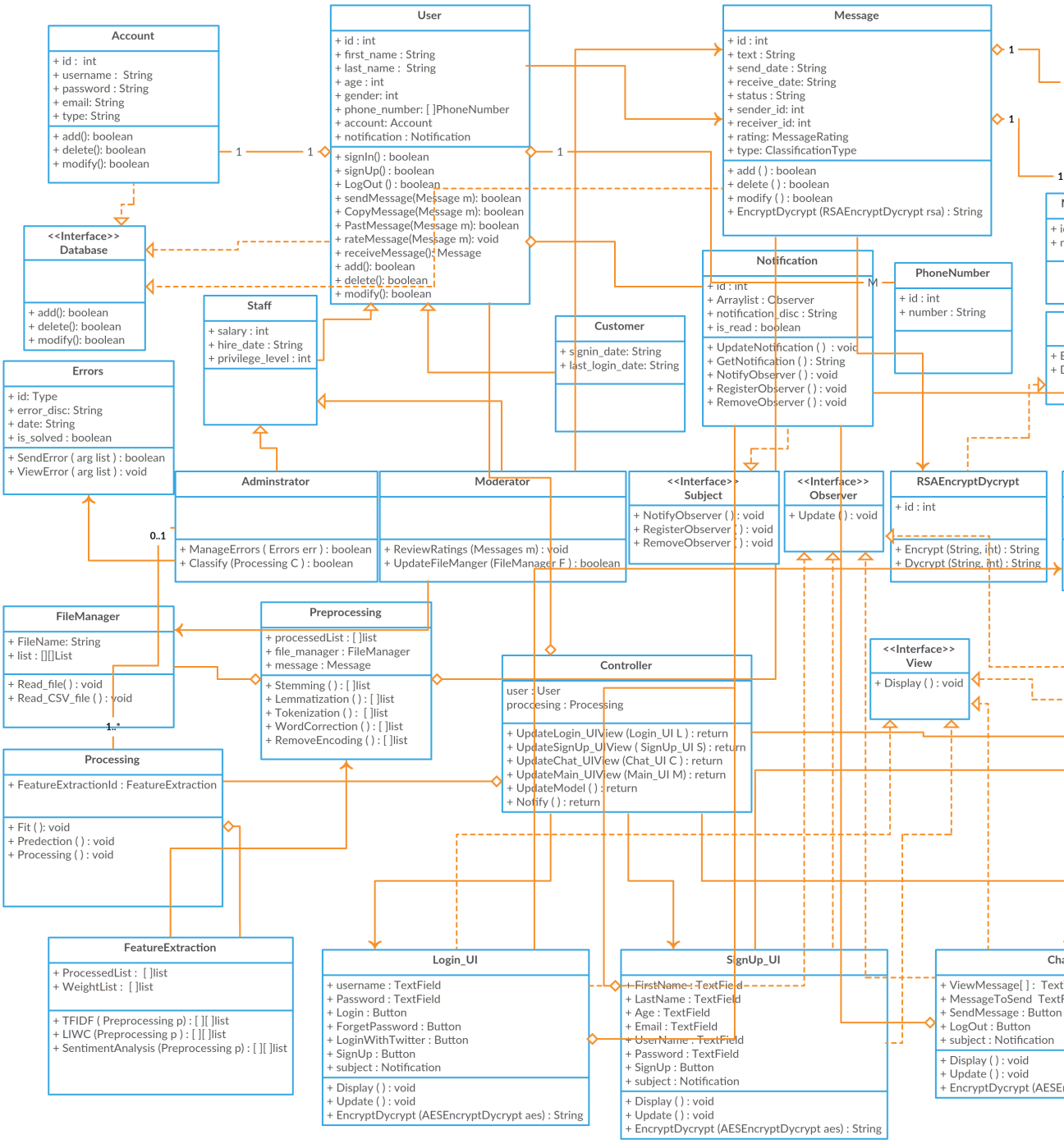


Figure 3.10: Class Diagram

3.8.2 Class descriptions

3.8.2.1 User Class

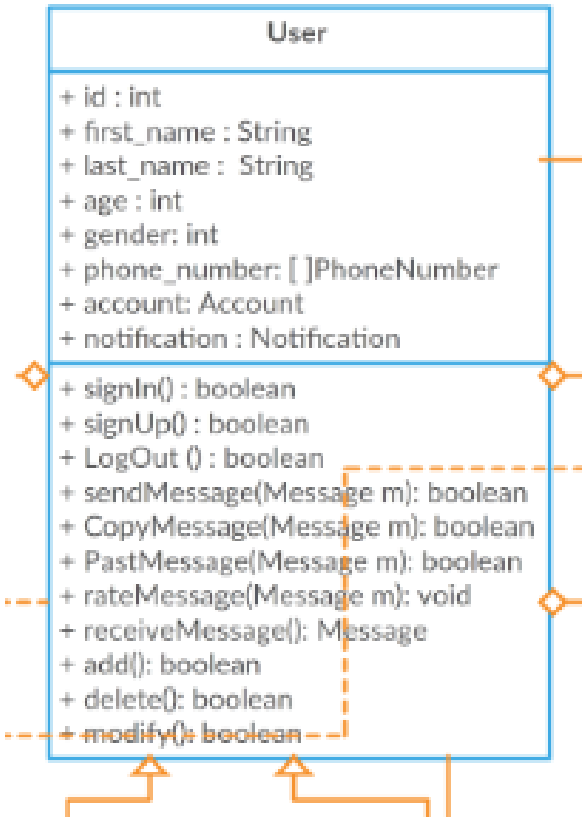


Figure 3.11: User class

Class name: User

List of super classes: N/A

List of sub classes: Administrator and customer.

Purpose: this class is used hold user info inside it

Collaborations: this class has account for security and has message as object in this class.

Attributes :

(a) int :id

(b) string : firstName

(c) string : lastName

(d) string : email

(e) account Account

(f) Int : Gender

(g) Int : Age

(h) Int : Gender

(i) Int : Age

(j) phonenumber[] : phonenumber

Operations :

(a) boolean sign in( ) : creates a session for the user to stay logged in through all pages

(c) void signUP ( ) : creates new user in the database

(d) boolean send message( ) : Send message to another user

(d) boolean Copy message( ) : Copy message to clipboard

(d) boolean Past message( ) : Past message from clipboard to UI

(e) Message Recicve Message( ) : Recive message from another user

(f) void Rating( ) : Rate the prectetage of bullying in mesaage from 1 to 5.

3.8.2.2 Customer Class

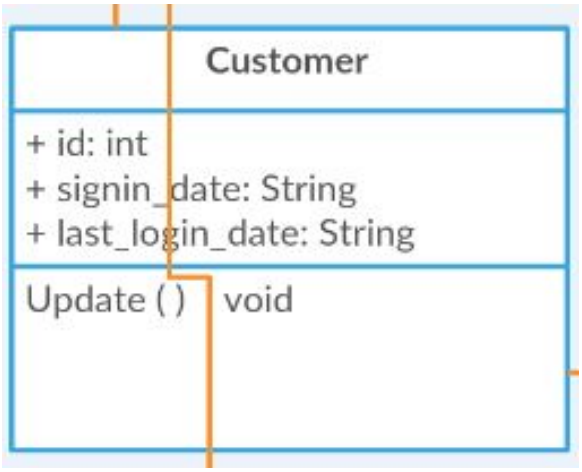


Figure 3.12: Customer class

Class name: Customer

List of super Classes :

(a) User

List of sub Classes : N/A

Purpose: this class is used hold Customer Date inside it

Attributes

(b) string: Sign in date

(C) string Last sign Up date

Operations :

3.8.2.3 Administrator Class

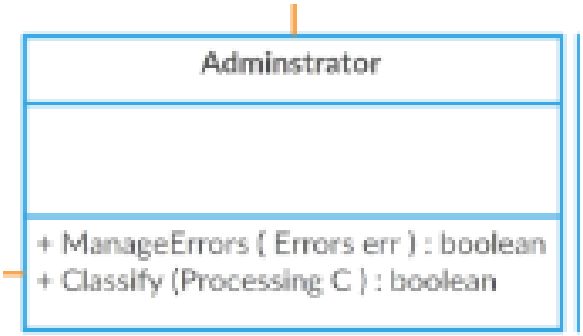


Figure 3.13: Adminestretor class

Class name: Administrator

List of super Classes :

(a) User

List of sub Classes : N/A

Purpose: this class is used to do some function for the administrator

Collaborations:

Attributes

Operations :

(a) Boolean ManageErrors(Errors err)

(b) Boolean Classify(Processing C)

3.8.2.4 Error Class

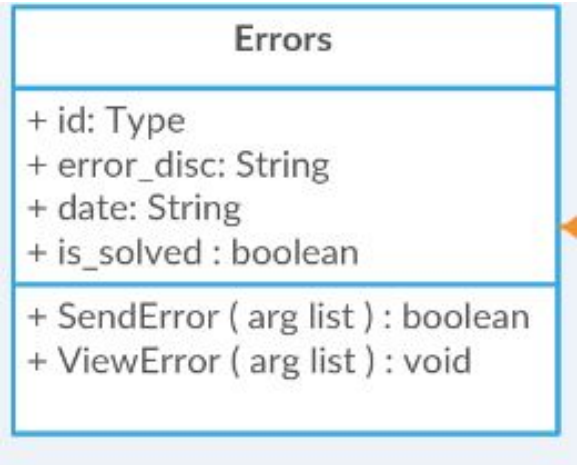


Figure 3.14: Error class

Class name: error

List of super Classes : N/A

List of sub Classes : N/A

Purpose: this class is used for viewing errors and sending them.

Collaborations: this class gives functions to the customer Class.

Attributes

(a) int : id

(b) string : Error Disc

(c) String: Date

(d) Boolean : is solved

Operations :

(a) Boolean sendError( ) : Send Error to the admin

(b) Void View Error() : View Error to user.

3.8.2.5 Notification Class

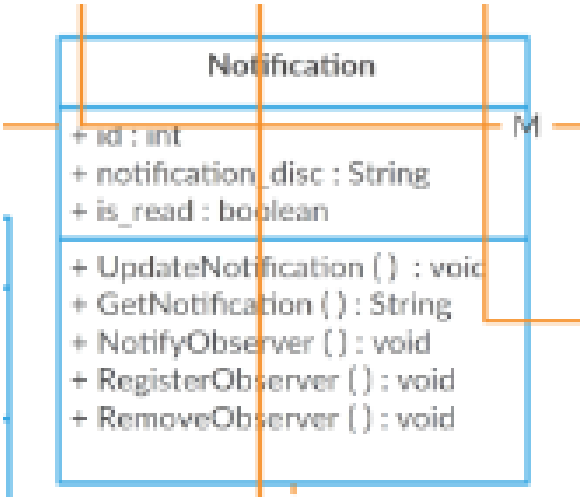


Figure 3.15: Notification class

Class name: notification

List of super Classes : N/A

List of sub Classes: N/A

Purpose: Sending notifications to the users.

Collaborations: this class will be aggregated from class user

Attributes

(a) int : id

(b) Arraylist : observers

(c) string : Notification Disc

(d) Boolean : is Read

Operations

(a)Notify observer( )

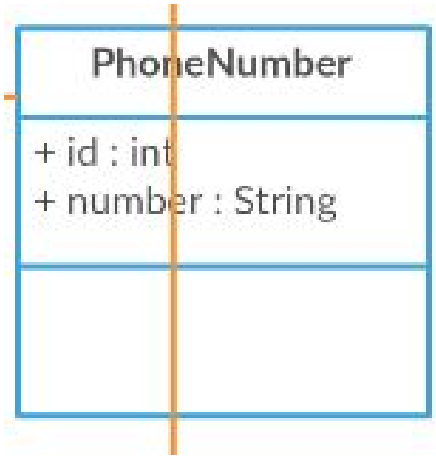
(b)Register observer()

(c)Remove observer()

(d)UpdateNotification()

(e)GetNotification()

3.8.2.6 Phone Number Class



number.jpg

Figure 3.16: Phone Number

Class name : PhoneNumber

List of Superclasses : User

List of Subclasses : N/A

Purpose : hold the numbers of all users

Collaborations : Inheritance from class User , Association with class Database

Attributes :

(a) int:id

(b) String:number

(c) int:UserId

Operations : N/A

Constraints : UserId is inherited from the User class properly as well as having the number correctly entered to be stored with no issues in the database

3.8.2.7 Processing class

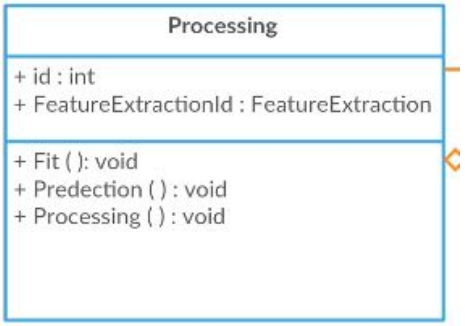


Figure 3.17: Processing class

Class name : Processing

List of Superclasses : N/A

List of Subclasses :

(a) FeatureExtraction

Purpose : First it holds the data being pushed from the FeatureExtraction class and then it's being classified as bullying or not

Collaborations :

(a) Aggregation from class FeatureExtraction

Attributes :

(a) int : id

(b) FeatureExtraction : FeatureExtractionId

Operations :

(a) Fit()

(b) Prediction()

(c) Processing()

Constraints : the array being pushed to the Processing class needs to be in correct values otherwise incorrect classification may occur.

3.8.2.8 File Manger Class

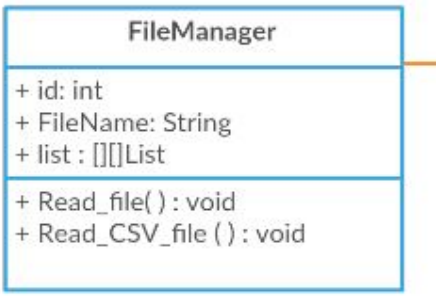


Figure 3.18: File Manager class

Class name : FileManager

List of Superclasses :

(a) Preprocessing

List of Subclasses : N/A

Purpose : deals with the dataset in order for it to be ready to be pushed to the Preprocessing class

Collaborations :

(a) Aggregates to Preprocessing

Attributes :

(a) int:id

(b) String:FileName

Operations :

(a) CS()

(b) readText()

Constraints : The dataset has to be modified correctly to be transferred correctly to the preprocessing class

3.8.2.9 Preprocessing Class

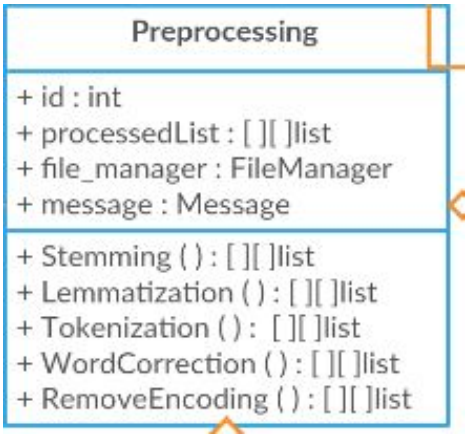


Figure 3.19: Preprocessing class

Class name : Preproceesing

List of Superclasses : N/A

List of Subclasses :

(a) FileManager

(b) FeatureExtraction

Purpose : the raw data being entered is being formatted and correctly analyzed and being prepared for feature extraction

Collaborations :

(a) Aggregation from FileManager

(b) Inheritance from FeatureExtraction

Attributes :

(a) int : id

(b) List : ProcessedList

(c) int : FileManagerId

Operations :

(a) Stemming()

(b) Lemmatization()

(c) Tokenization()

(d) WordCorrection()

(e) RemoveEncoding()

Constraints : data has to be passed on to the class with the correct formatting otherwise it would be useless from the beginning

3.8.2.10 Feature Extraction Class

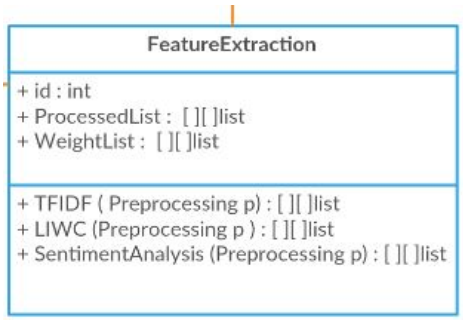


Figure 3.20: Feature Extraction class

Class name : FeaturExtraction

List of Superclasses :

(a) Preprocessing

List of Subclasses :

(a) Processing

Purpose : now that the data has been correctly formatted this class mostly deals with gathering information from the dull text and make it enriched with valuable meaning to be later passed on to the Processing phase

Collaborations :

(a) Inheritance from Preprocessing

(b) Aggregates to Processing

Attributes :

(a) int : id

(b) List : ProcessedList

(c) List : WeightList

Operations :

(a) TFIDF()

(b) LIWC()

(c) SentimentAnalysis()

Constraints : lists being passed from the preprocessing must be filled correctly and passed on with the correct formatting otherwise it would be difficult to extract meaningful values from the raw text being entered

3.8.2.11 Account Class

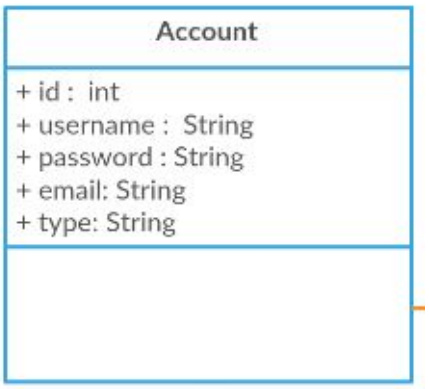


Figure 3.21: Account class

Class name : Account

List of Superclasses :

(a) User (b) Database List of Subclasses : N/A

Purpose : It holds the account details of every user.

Collaborations :

(a) Aggregation to class User

(b) Inhertance from class Databse

Attributes :

(a) int : id

(b) String : username

(c) String : password

(d) String : email

(e) String : type

Operations : N/A

Constraints : nn

3.8.2.12 Message Class

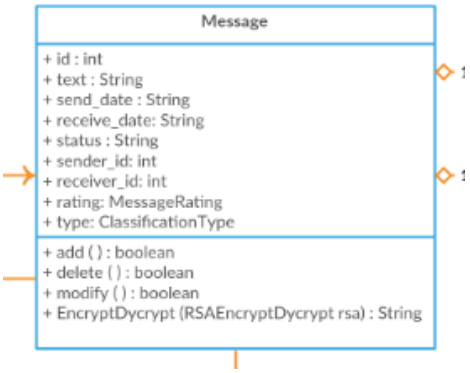


Figure 3.22: Message class

Class name : Message

List of Superclasses :

(a) Preprocessing

List of Subclasses :

(a) ClassificationType

(b) MessageRating

Collaborations :

(a) Aggregation from class Classification type

(b) Aggregation from class MessageRating

(c) Aggregation to class Preprocessing

Attributes :

(a) int : id

(b) String : text

(c) String : send-date

(d) String : recieve-data

(e) String : status

(f) int : sender-id

(g) int : reciever-id

(h) MessageRating : rating

(i) ClassificationType : type

Operations :

(a) boolean add ( )

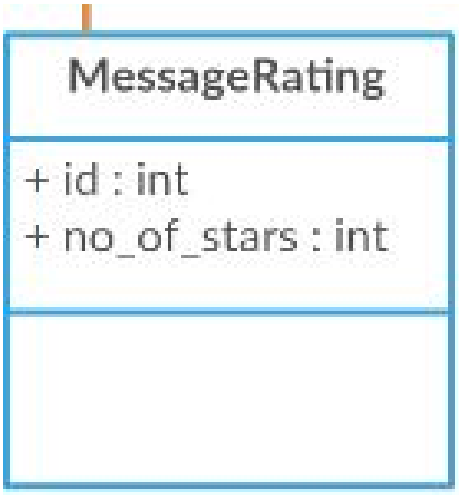
(b) boolean delete ( )

(c) boolean modify ( )

(d) String EncryptDycrypt (RSAEncryptDycrypt rsa)

Constraints : nn

3.8.2.13 Message Rating Class



rating.jpg

Figure 3.23: Message Rating class

Class name : MessageRating

List of Superclasses :

(a) Message

List of Subclasses : N/A

Collaborations :

(a) Aggregation to class Message

Attributes :

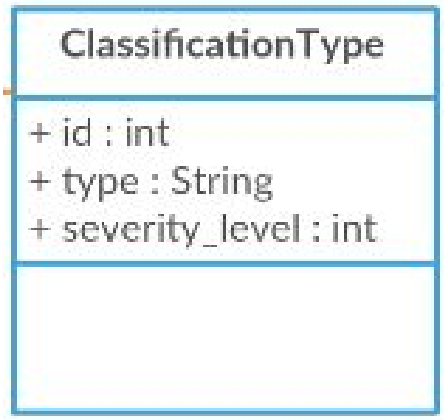
(a) int : id

(b) int : no-of-stars

Operations : N/A

Constraints : nn

3.8.2.14 Classification Type Class



type.jpg

Figure 3.24: Classification Type class

Class name : ClassificationType

List of Superclasses :

(a) Message

List of Subclasses : N/A

Collaborations :

(a) Aggregation to class Message

Attributes :

(a) int : id

(b) String : type

(c) int : severity-level

Operations : N/A

Constraints : nn

3.8.2.15 Login_UI Class

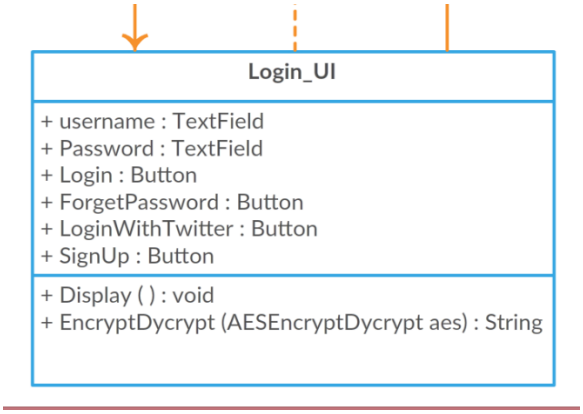


Figure 3.25: Login UI

1-Class name : Login_UI

2-List of Superclasses : View, EncrypteDycrypte

3-List of Subclasses : None

4-Purpose : This class is used to login to your account

5-Collaborations : This class implements "View" interface to be able to draw all the views on the UI and it implements "EncrypteDycrypte" to encrypte and dycrypte Login information

6-Attributes :

(a)TextField: username

(b)TextField: Password

(c)Button: Login

(d)Button: ForgetPassword

(e)Button: LoginWithTwitter

(f)Button: SignUp

(g)Notification: subject

7-Operations :

(a) String EncryptDycrypt (AESEncryptDycrypt aes)

(b) void Display()

(C) void Update()

3.8.2.16 SignUp_UI Class

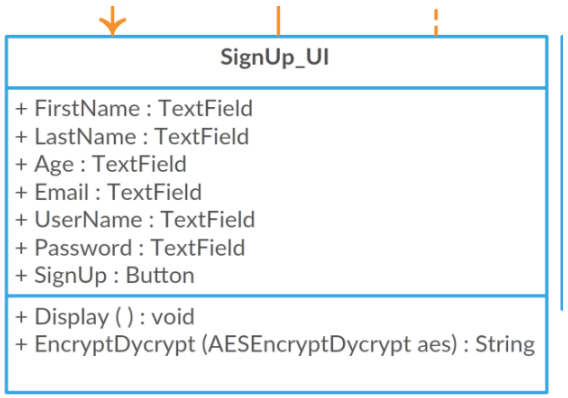


Figure 3.26: SignUp UI

1-Class name : SignUp_UI

2-List of Superclasses : View, EncrypteDycrypte

3-List of Subclasses : None

4-Purpose : This class is used to Create your new account

5-Collaborations : This class implements "View" interface to be able to draw all the views on the UI and it implements "EncrypteDycrypte" to encrypte and dycrypte signUp information

6-Attributes :

(a) TextField: FirstName

(b) TextField: LastName

(c) TextField: Age

(d) TextField: Email

(e) TextField: UserName

(f) TextField: Password

(g) Button: SignUp

(h) Notification: subject

7-Operations :

(a) String EncryptDycrypt (AESEncryptDycrypt aes)

(b) void Display()

(C) void Update()

3.8.2.17 Chat_UI Class

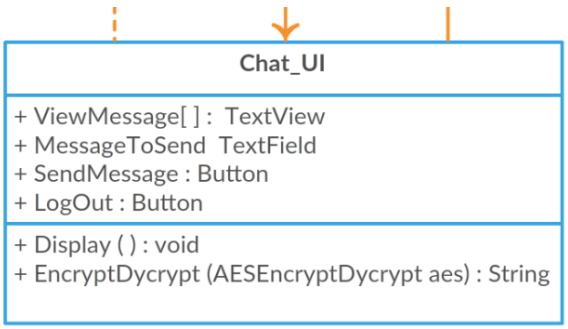


Figure 3.27: Chat UI

1-Class name : Chat_UI

2-List of Superclasses : View, EncrypteDycrypte

3-List of Subclasses : None

4-Purpose : This class have the chat messages between two users

5-Collaborations : This class implements "View" interface to be able to draw all the views on the UI and it implements "EncrypteDycrypte" to encrypte and dycrypte all messages send and received

6-Attributes :

(a)TextView: ViewMessage[]

(b)TextField: MessageToSend

(c)Button: SendMessage

(d)Button: LogOut

(e)Notification: subject

7-Operations :

(a) String EncryptDycrypt (AESEncryptDycrypt aes)

(b) void Display()

(C) void Update()

3.8.2.18 Main_UI Class

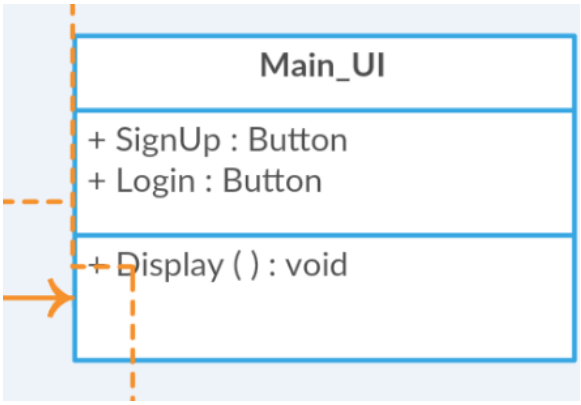


Figure 3.28: Main UI

1-Class name : Main_UI

2-List of Superclasses : View

3-List of Subclasses : None

4-Purpose : This class is where the user choose either to login or signup

5-Collaborations : This class implements "View" interface to be able to draw all the views on the UI

6-Attributes :

(a)Button: SignUp

(b)Button: Login

(c)Notification: subject

7-Operations :

(a) void Display()

(b) void Update()

3.8.2.19 Controller Class

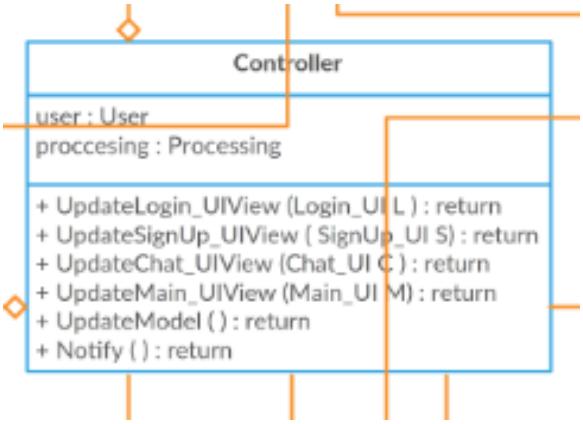


Figure 3.29: Controller

1-Class name : Controller

2-List of Superclasses : None

3-List of Subclasses : None

4-Purpose : This Class regulates the communication between models and views

5-Collaborations :

6-Attributes :

(a)User: user

(a)Processing: processing

7-Operations :

(a)void UpdateLogin_UI(Login_UI L)

(b)void UpdateSignUp_UI(SignUp_UI S)

(c)void UpdateChat_UI(Chat_UI C)

(d)void UpdateMain_UI(Main_UI M)

(e)void UpdateModel()

3.8.2.20 RSAEncryptDycrypt Class

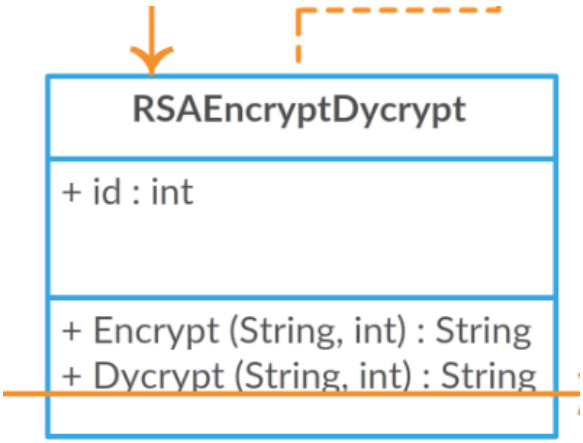


Figure 3.30: RSA

1-Class name : RSAEncryptDycrypt

2-List of Superclasses : EncryptDycrypt

3-List of Subclasses : None

4-Purpose :This class implements the RSA Encryption and Dycryption algorithm

5-Collaborations : The class implements "EncryptDycrypt" interface

6-Attributes :

(a)id : int

7-Operations :

(a)String Encrypt (String txt,int value)

(b)String Dycrypt (String txt,int value)

3.8.2.21 AESEncryptDycrypt Class

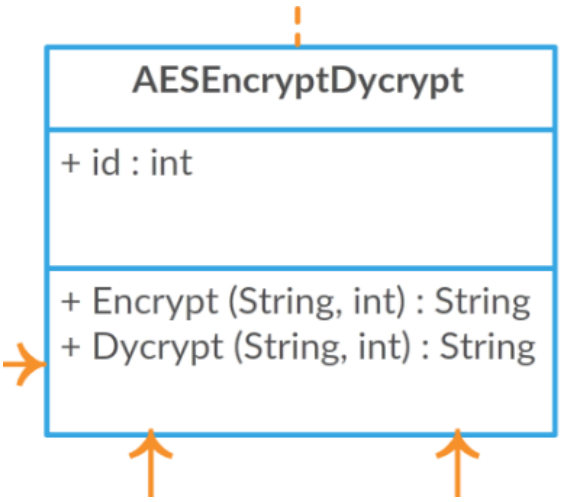


Figure 3.31: AES

1-Class name : AESEncryptDycrypt

2-List of Superclasses : EncryptDycrypt

3-List of Subclasses : None

4-Purpose : This class implements the AES Encryption and Dycryption algorithm

5-Collaborations : The class implements "EncryptDycrypt" interface

6-Attributes :

(a)id : int

7-Operations :

(a)String Encrypt (String txt,int value)

(b)String Dycrypt (String txt,int value)

3.8.2.22 Staff Class

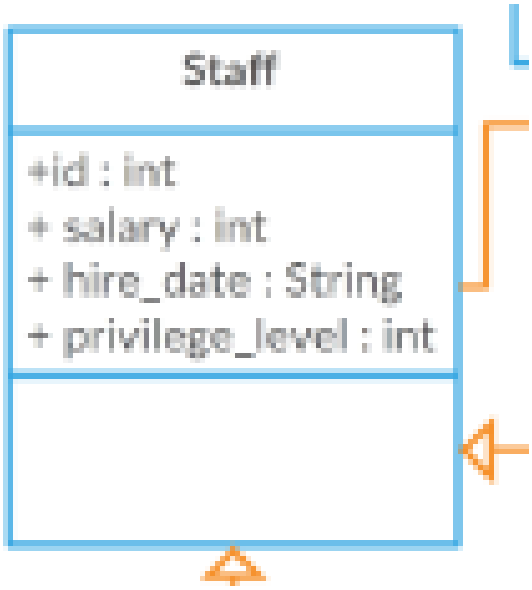


Figure 3.32: Staff

1-Class name : Staff

2-List of Superclasses : User

3-List of Subclasses : Administrator,Moderator

4-Purpose : This class holds the common information of working staff

5-Collaborations : The class extends "User" class

6-Attributes :

(a)salary : int

(b)hire_date : String

(c)privilege_level : int

7-Operations :

3.8.2.23 Moderator Class

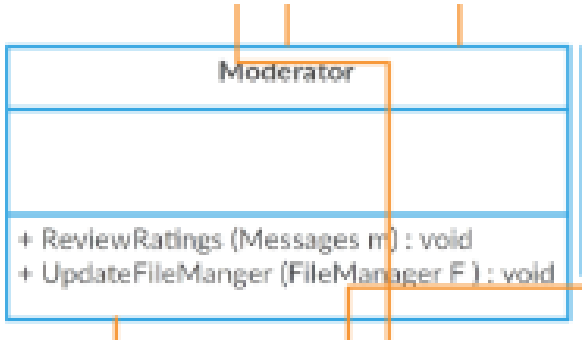


Figure 3.33: Moderator

1-Class name : Moderator

2-List of Superclasses : Staff,User

3-List of Subclasses :

4-Purpose : This class has some functions that Moderators use

5-Collaborations : The class extends "Staff" class

6-Attributes :

7-Operations :

(a) void ReviewRatings(Messages m)

(b) Boolean UpdateFileManger(FileManger F)

3.9 Operational Scenarios

3.9.1 Use Case

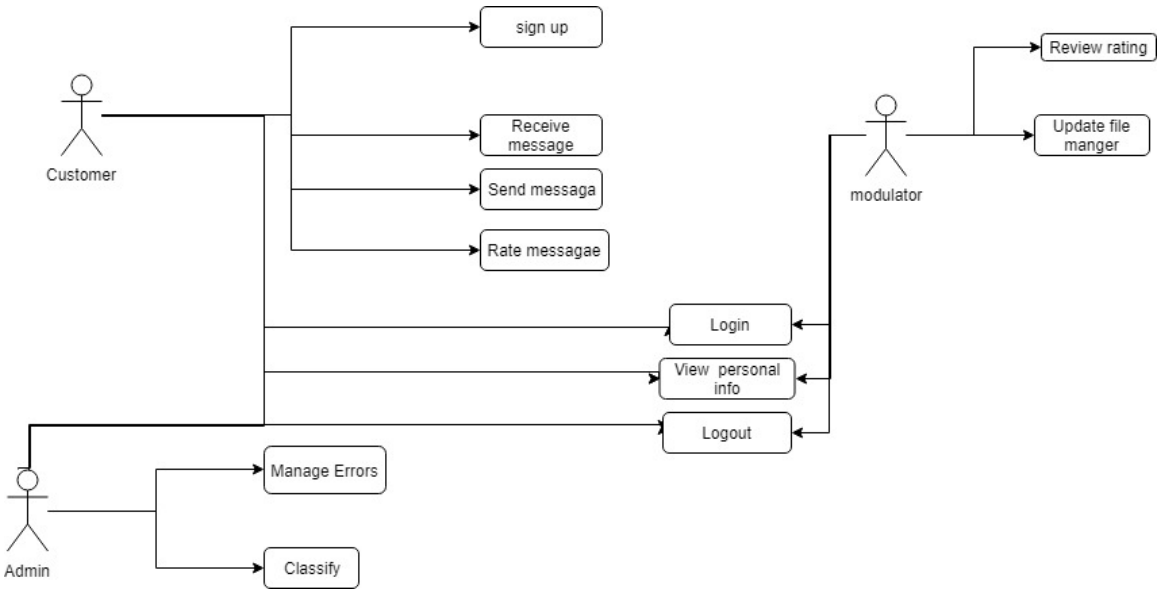


Figure 3.34: AES

Our operational scenario is as follows, we have a ready made dataset imported from kaggle website containing messages and their their weights, the dataset does appear to be labeled and supervised by the people who made it, first step is importing the dataset properly into our system and reading each message and value correctly in order for it to enter the preprocessing successfully, the dataset is now labeled and divided correctly in order for it to be entered to the feature extraction, in this phase meaningful values and expressions are extracted from each sentence and then passed on to the classifiers in order to come up with their classification whether it's cyber-bullying or not. Moving on to what will be implemented in the system. Which is having the user himself rate the messages that we have discovered as cyberbullying and then send that feedback to us for re-visioning, if its a match then we are doing something right, if not, then that message will be sent back once again from the developers back to the training phase with a new classification for it and adapting the results to meet the users intended results. Yes there may be some issues here and there when it comes to user miss usage but it's a part we are willing to handle with care.

3.10 Preliminary Schedule Adjusted

Task Name	Start Time	Finish
Idea Discussion	1/8/2018	1/8/2018
Idea Research	1/8/2018	13/9/2018
Proposal Writing	13/9/2018	16/9/2018
Implementing Prototype	16/9/2018	17/9/2018
Delivering Rehearsal	18/9/2018	18/9/2018
Delivering Proposal	18/9/2018	26/9/2018
Doing Survey	10/10/2018	20/10/2018
Implementing Demo	20/10/2018	25/10/2018
Writing SRS	25/10/2018	30/10/2018
Training Model	30/10/2018	25/11/2018
Preparing For External Examiner	25/11/2018	3/12/2018
Building Desktop App	3/12/2018	18/1/2019
Writing SDD	18/1/2019	1/2/2019
Building Android App	1/2/2019	1/4/2019
Preparing For Implementation Evaluation	1/4/2019	25/4/2019
Writing 8 Pages Paper	25/4/2019	28/4/2019
Testing and Debuging Project	28/4/2019	7/5/2019
Writing Final Thesis	10/5/2019	25/5/2019
Presenting Final Thesis	25/6/2019	25/6/2019

3.11 Preliminary Budget Adjusted

3.11.0.0.1 Google Machine Learning engine 0.7usd/h

3.12 Appendices

3.12.1 Definitions, Acronyms, Abbreviations

Software delivery lifecycle (SDLC)

3.12.2 Collected material

N/A

Chapter 4

System Design Document

4.1 Introduction

4.1.1 Purpose

In this document we're stating the main (non)functional requirements as well as targeting our main audience This software design document describes the architecture and system design of SSM. Our audience are the social media platforms e.g. Facebook, Twitter, etc.

4.1.2 Scope

Social Media platforms will benefit from our software as cyberbullying rates will drop. This application will be held by July, 2019 and will cost zero pounds; since its done for academic purposes.

4.1.3 Overview

Our document is organized as follows: System Overview, System Architecture, Data Design, Component Design, Human Interface Design, and Requirements Matrix. Our software will work on reducing the cyberbullying rates and make sure the social media is a safe place for everyone to feel free and express their words freely without humiliating other people.

4.1.4 Reference Material

1) Cyberbullying System Detection and Analysis
2) Experts and Machines Against Bullies A Hybrid Approach to Detect Cyberbullies

3) Cyberbullying Detection using Time Series Modeling

4) Machine Learning Approach for Detection of Cyber-Aggressive Comments by Peers on Social Media Network

5) SDD-HCI- Professor Doctor Ayman Ezzat

4.1.5 Definitions and Acronyms

Term	Definition
Software Design Document (SDD)	Used as the primary medium for communicating software design information.
SSM	Safe Social Media.
Activity Diagram	is a graphical representation of an executed set of procedural system activities and considered a state chart diagram variation.
Sequence Diagram	represents object collaboration and is used to define event sequences between objects for a certain outcome.
SVM	Support Vector Machine.
Design Entity	An element of a design that is structurally and functionally distinct from other elements.

4.2 System Overview

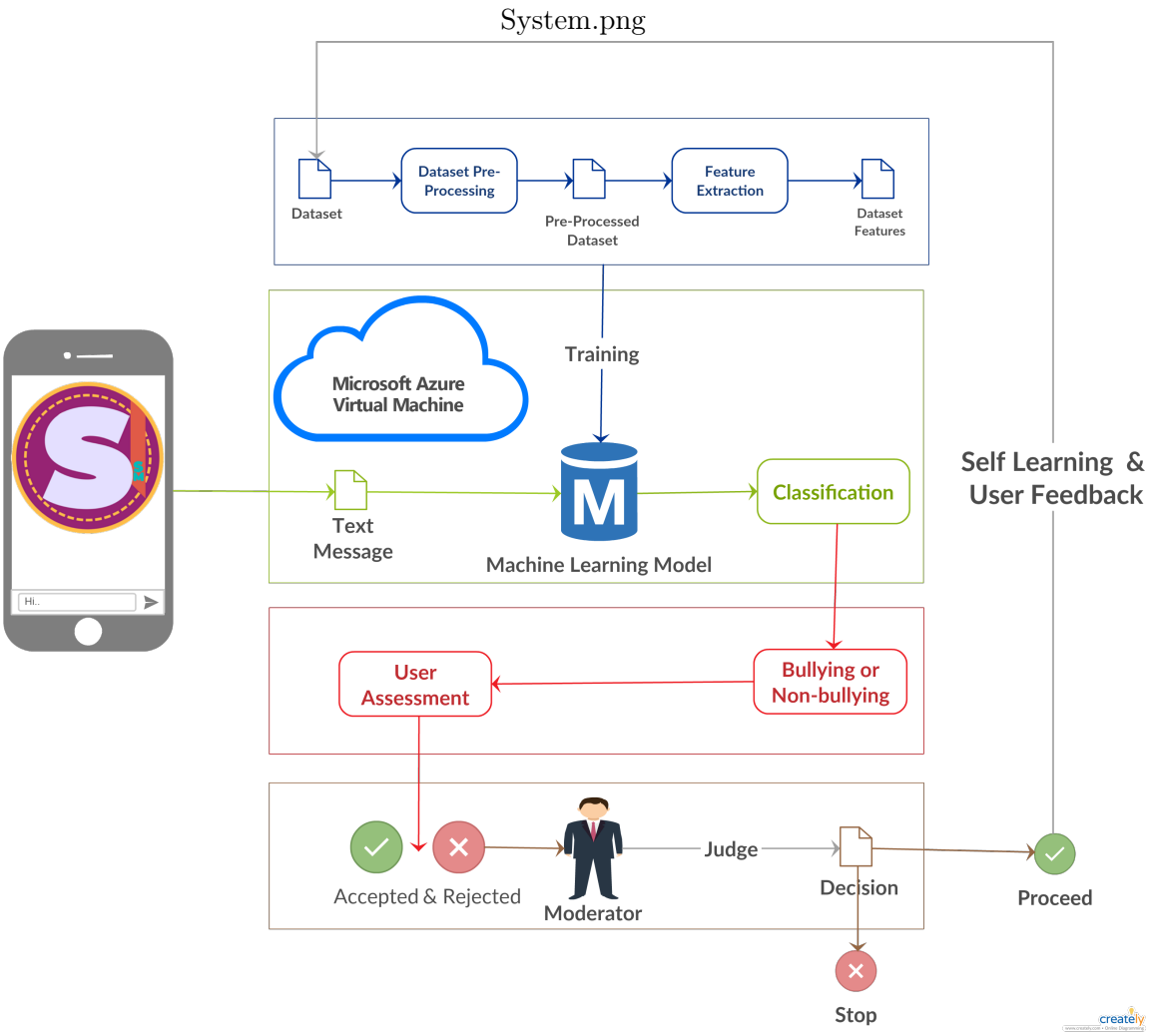


Figure 4.1: System overview

In order to reach high accuracy in detecting cyberbullying. We developed a system that consists of five stages.

1)Pre-processing

2)Feature Extraction that consists of three stages

3)Training and testing

4)Classifying

5)Self learning

4.2.1 Pre-processing

In this stage we have six phases

1) Tokenize every word in the text

2) Lower the letters

3) Correct the words that are incorrect

4) Put every word to its root

5) Remove any code in the text

6) Remove the stop words.

4.2.2 Feature extraction

We are using TFIDF and sentiment analysis to decrease the false positive instances in the detection of cyberbullying.

4.2.3 Training and testing

We are using the results of our Feature extraction to train our model and then perform prediction into untested corpus.

4.2.4 Classification

When the Application sends the text to the SVM classifier, then the classifier will send the result of the classification if it is cyberbullying or not.

4.2.5 Self-learning

Both accepted and rejected assessments by the user will be taken into consideration but not all of them will be taken once more to our dataset to complete the process of self-learning.

4.3 System Architecture

4.3.1 Architectural Design

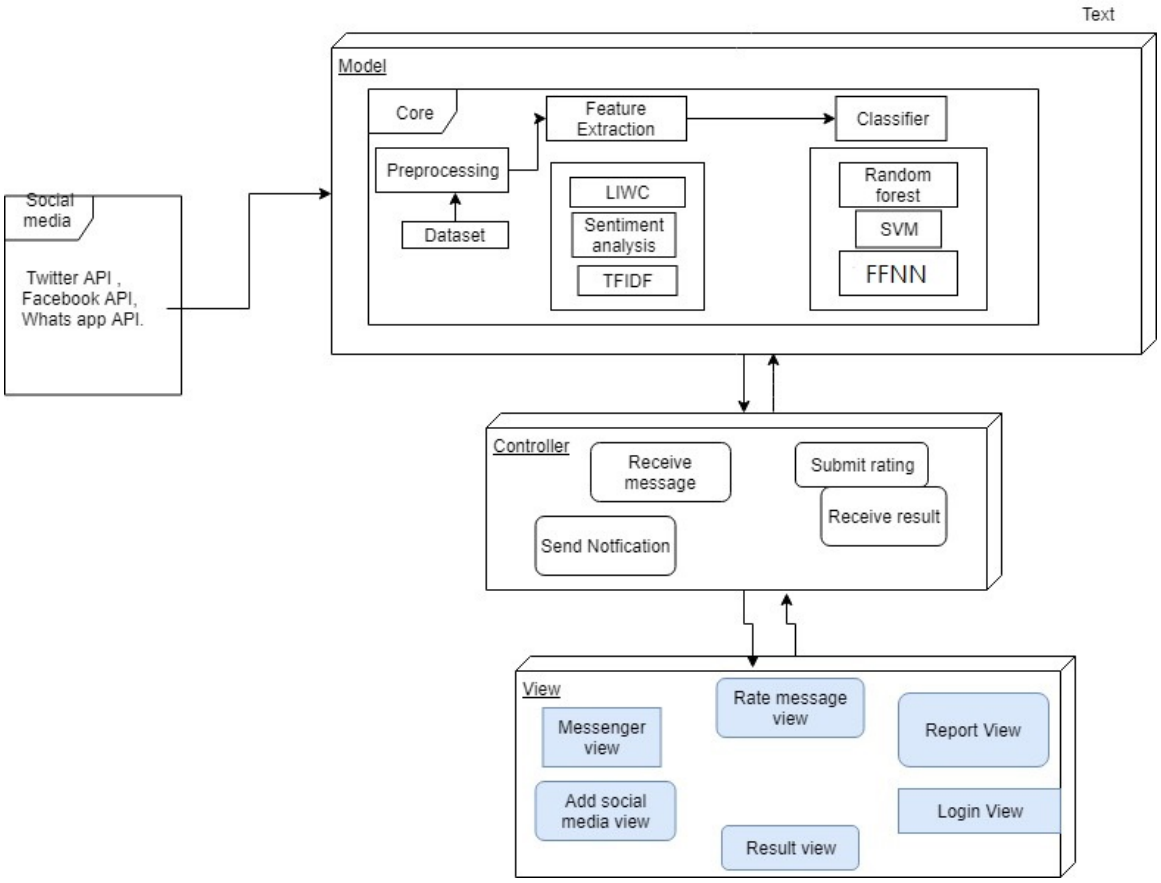


Figure 4.2: System Architecture

4.3.1.1 Model

Pre-Processing: it is the first module that consists of 4 parts

1-Toknization: the first phase that separate each word in the dataset to differentiate between significant words and non-significant words.
2-Stop Words Removal: Natural language toolkit to detect and remove the non-significant words like (and, a, an).
3-Stemming: a process that return every word to its root word.

4-Word Correction :it takes the word and correct the spelling mistakes in it using Bing API.

Feature Extraction: features are extracted from the document before feeding in the classifier and it is consists of two parts.

1-TFIDF: Short for term frequency inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document.
2-Sentiment Analysis: The process of computationally identifying and categorizing opinions expressed in a piece of text, especially in order to determine whether the writer's attitude towards a particular topic, product, etc.: is positive, negative, or neutral.

Classification: In this part the extracted features are sent to classifiers for classification. 1-Support-Vector Machines (SVMs, also support-vector networks): are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis.
2-FeedForward Neural Network: feedforward neural network is an artificial neural network wherein connections between the nodes do not form a cycle, also is one of the simplest neural network.

4.3.1.2 Controller

Controller acts on both model and view. It controls all data in the model and updates the view whenever any data change occurs. It is responsible for keeping the model separated from the views section. It handles all the data produced from the Pre-processing, feature extraction, and classification then updating the interfaces takes place.

4.3.1.3 View

This section represents the user mobile interfaces loaded by the system and providing the a chatting application the detect cyberbullying in the chat then it will make the user rate the classification.

Rating View: This view will allow the to assess and sent his feedback whether it is acceptance or rejection.

Login view : It will allow the user login with his user name and password.

Add social media view: Will allow the user to add whatsapp, messenger, or twitter to our app.
Messenger view: Will allow to chat in this view.
Result view: Will the show the user the result of the classification of message.

4.3.2 Decomposition description

4.3.2.1 Class diagram

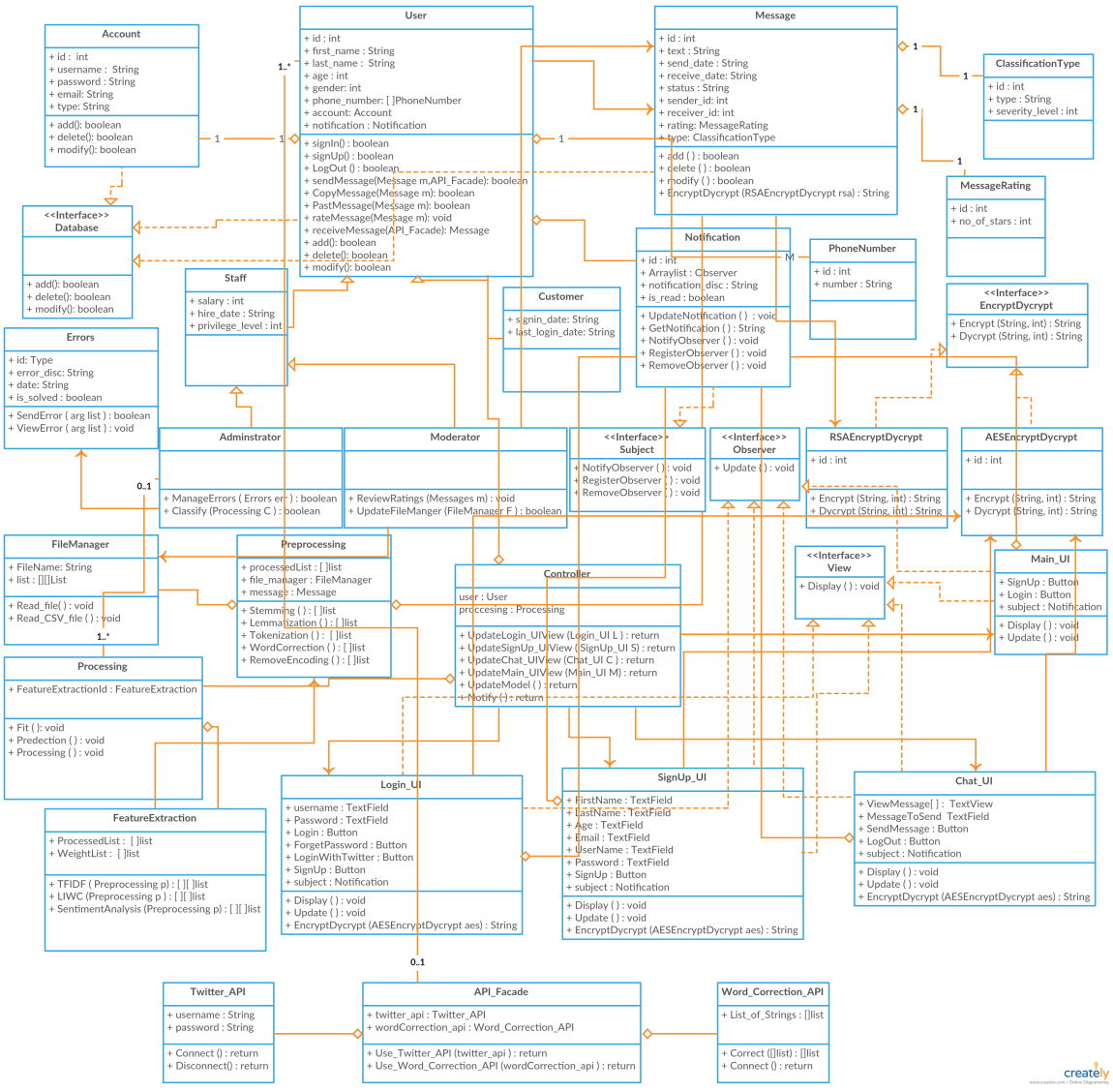


Figure 4.3: Class diagram

4.3.2.2 MVC Design Pattern

We are using MVC to make every part of the system at it is own and that helps for creating many views as we want to make mobile application and desktop application.

4.3.2.3 Facade Design Pattern

Facade Class is associating the classes the contain the twitter API and bing API the the user will associate the facade class.

4.3.2.4 Strategy Design Pattern

We use the strategy design pattern because we are using two Encryption/decryption methods. We have interface class and two concrete classes inherit from the interface class.

4.3.2.5 Observer Design Pattern

A software design pattern in which an object maintains a list of its dependents and notifies them automatically of any state changes. (Notification Class)

4.3.2.6 Activity Diagram

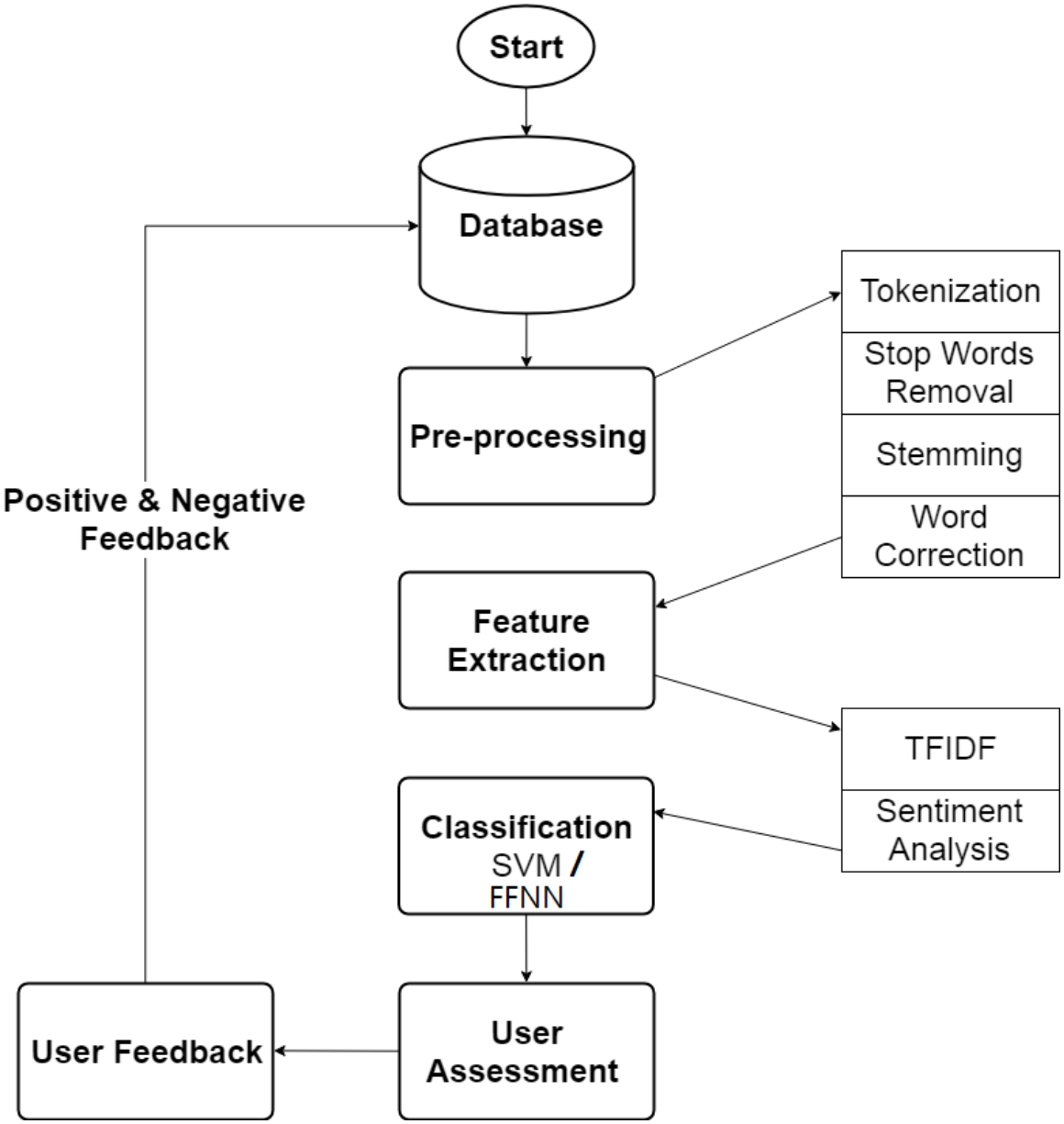


Figure 4.4: Activity Diagram

4.3.2.7 Sequence Diagram

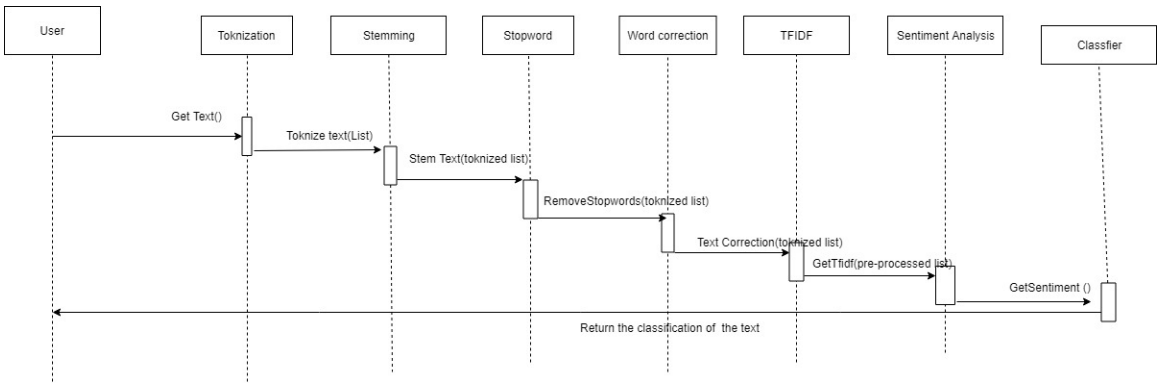


Figure 4.5: Sequence Diagram

In this diagram we are the showing the processes of text classification in the system and how are the stages related in the system.

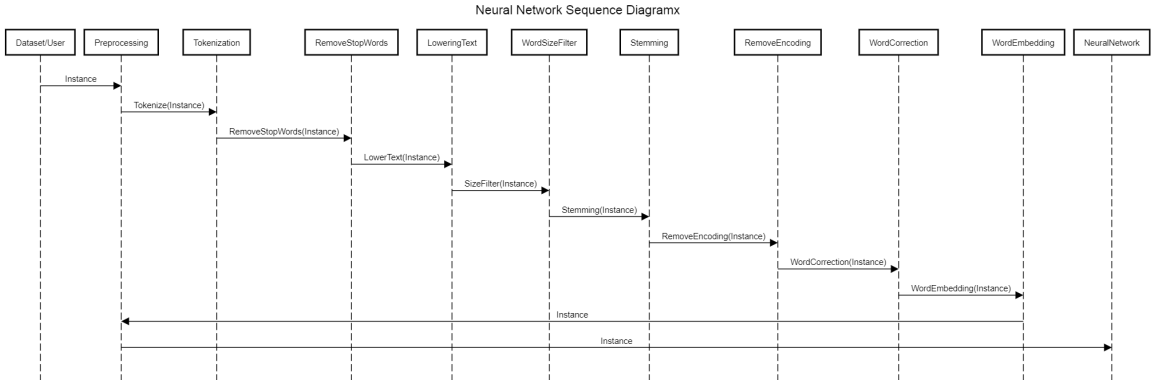


Figure 4.6: Neural Network Sequence diagram

4.3.3 Design Rationale

The SSM is based in MVC design pattern, where the model (Cloud) handles the back-end processing like feature extraction and classification, the controller takes the information from the model and deliver it to the view for the user interface, the view takes the event from the user using the event listeners and deliver it to the controller, then the controller gives it to the model for further processing. Using the MVC design pattern ensure the re-usability of the SSM for other projects, and the easy maintenance without going deep inside the core.

4.4 Data Design

4.4.1 Data Description

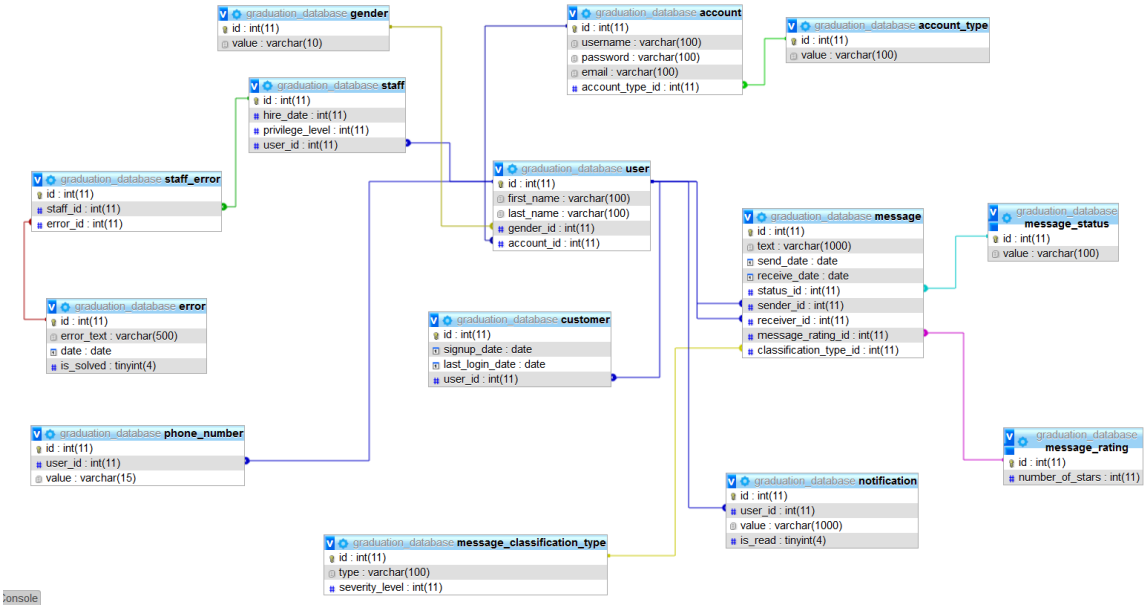


Figure 4.7: Database design

4.4.2 Data Dictionary

1- Account: This entity will hold the account information for every SSM user. It will store information like: username, password and email.

2- Account Type: This entity will hold the type of the account.

3- Customer:This entity will hold the account information of the user like: sign up date, last login date

4- Error: This entity will hold information about errors in the application.

5- Gender: This entity will hold the type of Gender like: male or female.

6- Message: This entity will hold the details of all send and received messages.

7- Message Classification Type: This entity will hold the result of classification of the received message.

8- Message Rating: This entity will hold the user rating of the received message.

9- Message Status: This entity will hold the Status of the message.

10-Notification: This entity will hold the user notification and it's details.

11- Phone Number: This entity will store all the phone number of all users.

12- Staff: This entity will hold the information of all staff like: hire date, privilege level.

13- Staff Error: This entity will hold the information of which error is assigned to which staff

14- User: This entity will hold the information of all users: customers and staff like: first name, last name

4.5 Component Design

4.5.1 Machine learning

4.5.1.1 TFIDF

TFIDF stands for term frequency inverse document frequency and is used to weight every word in the sentence to measure of how important that term is in the whole corpus.

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

Figure 4.8: TFDF equation

4.5.1.2 Sentiment Analysis

Sentiment analysis is used to know the polarity of the sentence to help in deciding whether it is cyberbullying or not.

4.5.1.3 SVM

We are using SVM(Support Vector Machine) for classification and we send the result of TFIDF and Sentiment analysis as parameters to the classifier.

$$\mathcal{D} = \left\{(\boldsymbol{x}_i, y_i) | \boldsymbol{x}_i \in \mathbb{R}^p, y_i \in \{-1, 1\}\right\}_{i=1}^{n}$$

Figure 4.9: SVM equation

4.5.2 Neural Network

4.5.2.1 ReLU

ReLU stands for Rectified Linear Unit, it is used in the second layer in the feed forward neural network. We use it to eliminate the negative values in the neural network.

$$f(x) = x^+ = \max(0, x)$$

Figure 4.10: ReLU equation

4.5.2.2 Segmoid

Segmoid is used in the third layer in the feed forward neural network, and it's used to get results from 0 to 1.

$$f(x) = \frac{1}{1 + e^{-x}}$$

Figure 4.11: Segmoid equation

4.6 Human Interface Design

4.6.1 Overview of User Interface

The User interface is going to be very easy to use. First the user is going to sign up for the application after signing up a screen with adding the social media platform that the user tends to use with the application. after that the user will see normal screen like the normal social media application. After that the user see normal messages but when a message is classified as bullying it will appear blurred to the user with a note that this message is bullying and have the choice to show it and after showing it the user will be able to rate the classification results.

4.6.2 Screen Images

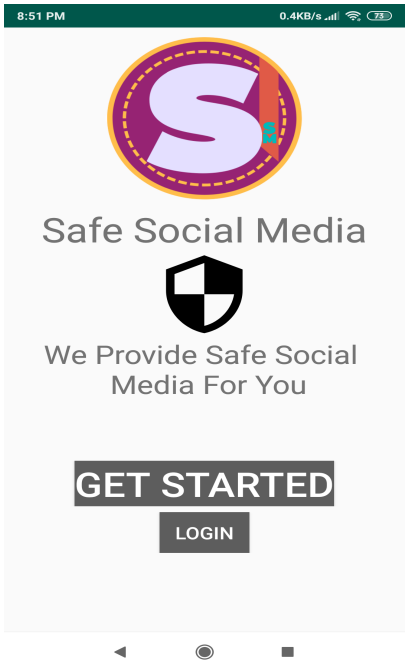


Figure 4.12: Rating

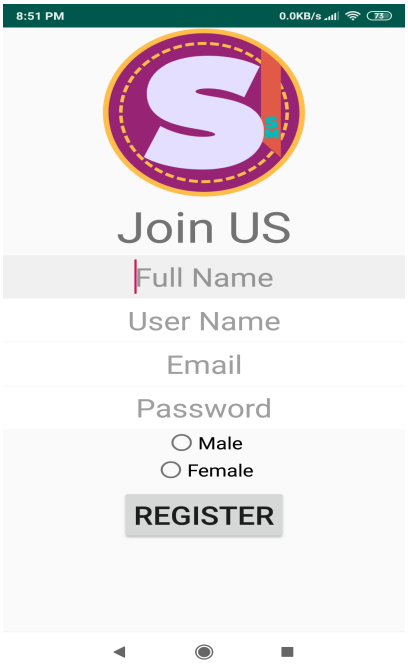


Figure 4.13: Register

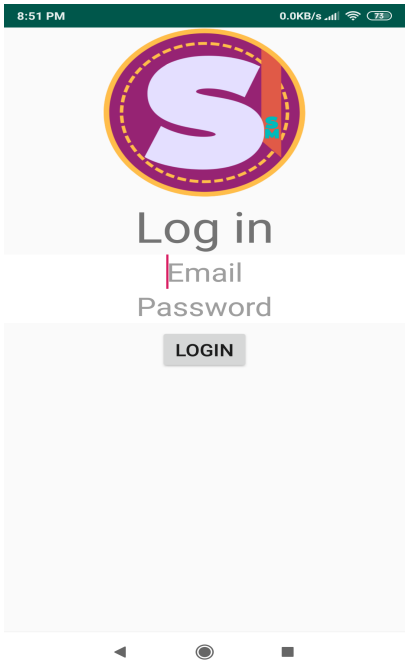


Figure 4.14: Log in

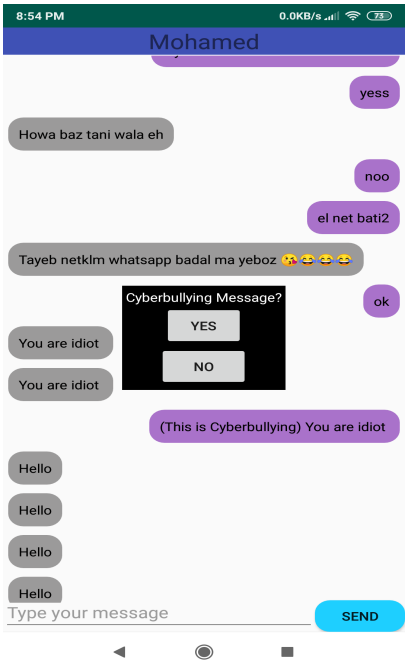


Figure 4.15: Rating

4.6.3 Screen Objects and Actions

- Figure 12 (Home): This is the main screen when the user open the application
- Figure 13 (Register): In this screen the user Register Account within our service
- Figure 14 (Log in): In this screen the user log in in his account within our service
- Figure 15 (Add Social Media): In this screen the user give our service access to his account on social media in order to monitor the messages
- Figure 16 (Rating): In this screen the user will be able to rate the classification results

4.7 Requirements Matrix

Requirement id	Requirement type	Description	Test strategy
3.1 Login	Required	Allow the user to login into the application	Only accept Entering valid user name and password
3.2 Signup	Required	Allow user to enter his information	Accept only the enters on the fields
3.3 adding social media	Required	Allow user to enter his social media user name and password	Must return acceptance to the user name and password
3.4 chatting menu	Required	Allow the user to see every chat he is doing	Must return all the chats he is doing
3.5 send message	Required	Allow the user to send message to another user	The message must contain at least one character
3.6 Show message	Required	Show all chatting between the users	Must show the messages between the users
3.6 Detect messages	Required	Send the received message to the user and return the result of the detection if yes blur the text	Must blur the detected Text
3.7 Rate the detected text	Required	Allow the user to rate the detection of the cyberbullying	Give the user the ability to rate the detection
3.8 Modulator rating message	Required	Allow the modulator to rate the message that rated as wrong detection	Take the rating from the modulator
3.9 Receive the message	Required	Receive the message from the social media application	Must be sure that message is received
4.0 Logout	Required	Take the user out of the application	End the session of the user

Chapter 5

Evaluation

5.1 Introduction

After making our purposed system, the system was passed through multiple experiments to test and identify the points of strength of the system proposed. The first experiment was mainly to recognize which classifiers are better with our dataset and compare there accuracy with the rest of the classifiers on the same dataset. The second experiment aims to compare the results of our purposed system with the other related work of cyberbullying detection. Finally, the third experiment goal was to get the users feedback after implementing the self learning module to compare it with the results before implementing our functions model.

5.2 Experiment 1 Cyberbullying detection classification

5.2.1 Goal

Decide which classifier gives the system the highest accuracy with cyberbullying detection.

5.2.2 Classifiers tested

1-SVM
2-Logistic regression
3-Random Forest
4-Naive Bayes

5-Feed Forward Neural Network

5.2.3 Task

First, we read the dateset and execute dataset balancing and filtering to remove noise data. Second, make do preprocessing filtered data. Finally, we extract features using Sentiment analysis and TFIDF then we feed it into the classifier.

5.2.4 Results

Classifier	SVM	FFNeural Network	Logistic regression	Naive Bayes	Random Forest
Accuracy	89.87	91.76	84.75	78.04	71.43
Precision	89.6	92.4	80.39	84	93.47
Recall	90.1	91.7	95	71.43	49.42

As we can see the difference in accuracy, precision and recall between FF Neural Network and SVM with the other classifiers is too large so we did apply SVM and FF Neural Network in our application.

5.3 Experiment 2 Comparing proposed system with related work

5.3.1 Goal

our goal is to compare the results of our proposed system that used SVM and FF Neural Network for classification with the work of [4]. In this work, they used logistic regression and SVM for classification and used the same data.

5.3.2 Task

we have calculated the average accuracy, recall, precision and F-score of our two classifiers. The summary of results is shown in table 5.1. To compare the work, it is found that our proposed NN model outperforms all other classifiers and is ranked as the best results in terms of average accuracy and F-Score achieving accuracy 91.76% and f-score 91.9%. In fig. 5.3 we are comparing between our best classifier with their best classifier in case of accuracy.

Finally, here in fig. 5.2 we are comparing between our best classifier with their best classifier in case of F-Measure.

5.3.3 Results

Table 5.1: COMPARISON WITH RELATED WORK

	Classifier	Avg. Accuracy	Avg. Recall	Avg. Precision	Avg. F-Score
Vikas S Chavan	Logistic regression	73.76	61.47%	64.4%	62.9%
	SVM	77.65%	58.29%	70.29%	63.7%
Current Results	Neural Network	**91.76%**	**91.7%**	**92.4%**	**91.9%**
	SVM	89.87%	90.1%	89.6%	89.8%

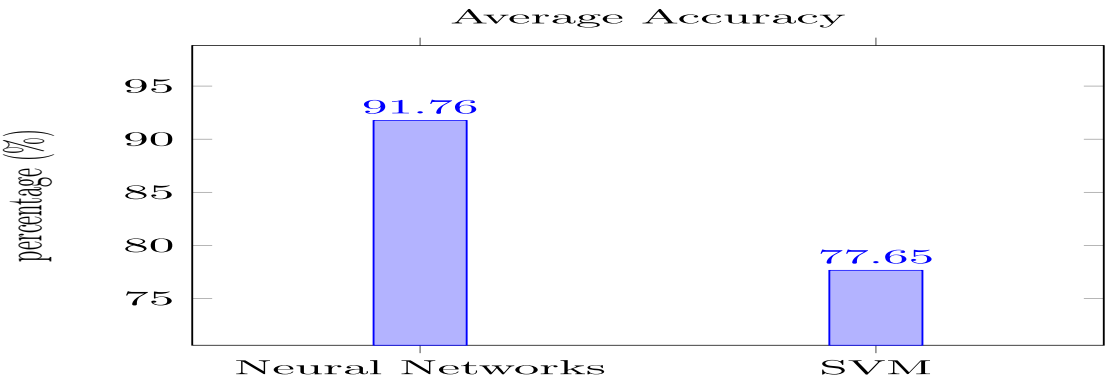


Figure 5.1: COMPARISON BETWEEN THE BEST CLASSIFIERS IN TERMS OF ACCURACY

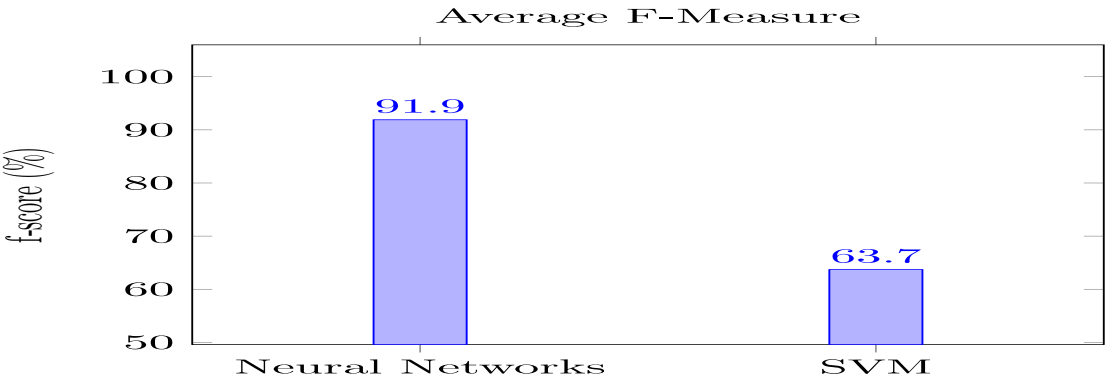


Figure 5.2: COMPARISON BETWEEN THE BEST CLASSIFIERS IN TERMS OF F-MEASURE

5.4 Compare Between Human and our classifier in the detection of cyberbullying

5.4.1 Goal

Our goal is to Decide how the classifier of our proposed system performs compared to manual classification by users.

5.4.2 Task

We gathered ten users each one was given ten sentences to decide which of them is cyberbullying Knowing what is cyberbullying and what is not. Then we feed these 10 sentences to the classifier to predict there class. Then we calculate the average between the users and then compare the result to our classifier's prediction. As we can see in below figure the comparison of the results.

5.4.3 Results

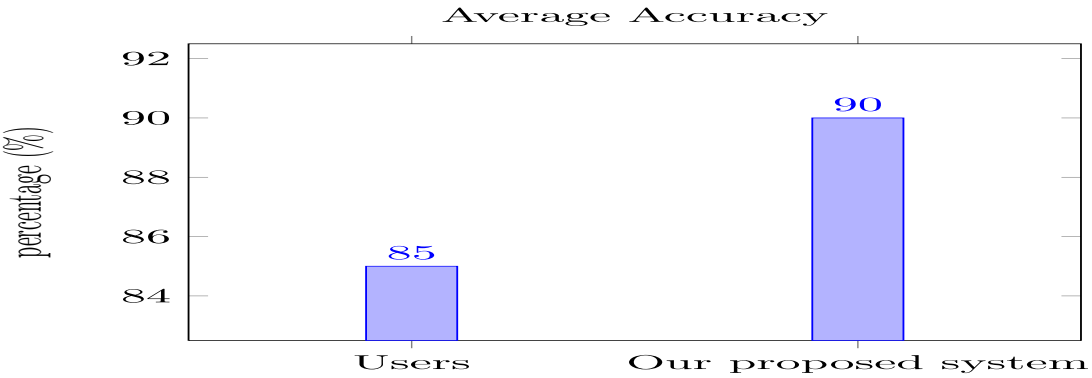


Figure 5.3: COMPARISON BETWEEN USERS and OUR SYSTEM IN TERMS OF ACCURACY

Our classifier results are higher than Human in this comparison because users usually consider if a sentence was sent to a close friend even if it contains cyberbullying, so they rated it as non-cyberbullying.

Chapter 6

Conclusion

One of the main challenges we faced while developing our purposed system that detects cyberbullying is false positive classification. Our goal is to reduce the false positive and increase the accuracy of the SSM system. We evaluated our system on two classifiers SVM and Neural Network and we used TFIDF and sentiment analysis algorithms for features extraction. The classifications were evaluated on different n-gram language models. We achieved 92.8% accuracy using Neural Network with 3-grams and 90.3% accuracy using SVM with 4-grams while using both TFIDF and sentiment analysis together. We found that our Neural Network performed better than the SVM classifier as it also achieves average f-score 91.9% while the SVM achieves average f-score 89.8%. By achieving this accuracy, our work is definitely going to improve cyberbullying detection to help people to use social media safely. However, detecting cyberbullying pattern is limited by the size of training data. Thus, a larger cyberbullying data is needed to improve the performance. Hence, deep learning techniques will be suitable in the larger data as they are proven to outperform machine learning approaches over larger size data.

6.1 Future directions

Our future work is to work on Sarcasm detection as it's an important field. Moreover, this work will help improve the accuracy of cyberbullying detection, as it will help differentiate between harassment comments and sarcastic comments.

Bibliography

[1] S. K. Bharti, R. Naidu, and K. S. Babu, "Hyperbolic feature-based sarcasm detection in tweets: A machine learning approach," in *2017 14th IEEE India Council International Conference (INDICON)*. IEEE, 2017, pp. 1–6.

[2] S. Bharti, B. Vachha, R. Pradhan, K. S. Babu, and S. Jena, "Sarcastic sentiment detection in tweets streamed in real time: a big data approach," *Digital Communications and Networks*, vol. 2, no. 3, pp. 108–121, 2016.

[3] M. Bouazizi and T. O. Ohtsuki, "A pattern-based approach for sarcasm detection on twitter," *IEEE Access*, vol. 4, pp. 5477–5488, 2016.

[4] V. S. Chavan and S. Shylaja, "Machine learning approach for detection of cyber-aggressive comments by peers on social media network," in *Advances in computing, communications and informatics (ICACCI), 2015 International Conference on*. IEEE, 2015, pp. 2354–2358.

[5] Y. Chen, Y. Zhou, S. Zhu, and H. Xu, "Detecting offensive language in social media to protect adolescent online safety," in *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom)*. IEEE, 2012, pp. 71–80.

[6] M. Dadvar and F. De Jong, "Cyberbullying detection: a step toward a safer internet yard," in *Proceedings of the 21st International Conference on World Wide Web*. ACM, 2012, pp. 121–126.

[7] M. Dadvar, D. Trieschnigg, and F. de Jong, "Experts and machines against bullies: A hybrid approach to detect cyberbullies," in *Canadian Conference on Artificial Intelligence*. Springer, 2014, pp. 275–281.

[8] M. Dadvar, D. Trieschnigg, R. Ordelman, and F. de Jong, "Improving cyberbullying detection with user context," in *European Conference on Information Retrieval*. Springer, 2013, pp. 693–696.

[9] H. Dani, J. Li, and H. Liu, "Sentiment informed cyberbullying detection in social media," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2017, pp. 52–67.

[10] P. Dharwal, T. Choudhury, R. Mittal, and P. Kumar, "Automatic sarcasm detection using feature selection," in *2017 3rd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*. IEEE, 2017, pp. 29–34.

[11] S. M. Isa, L. Ashianti *et al.*, "Cyberbullying classification using text mining," in *Informatics and Computational Sciences (ICICoS), 2017 1st International Conference on*. IEEE, 2017, pp. 241–246.

[12] H. H. S. Li, Z. Yang, Q. Lv, R. I. R. R. Han, and S. Mishra, "A comparison of common users across instagram and ask. fm to better understand cyberbullying," in *Big Data and Cloud Computing (BdCloud), 2014 IEEE Fourth International Conference on*. IEEE, 2014, pp. 355–362.

[13] E. Lunando and A. Purwarianti, "Indonesian social media sentiment analysis with sarcasm detection," in *Advanced Computer Science and Information Systems (ICACSIS), 2013 International Conference on*. IEEE, 2013, pp. 195–198.

[14] S. Murnion, W. J. Buchanan, A. Smales, and G. Russell, "Machine learning and semantic analysis of in-game chat for cyberbullying," *Computers & Security*, vol. 76, pp. 197–213, 2018.

[15] V. Nahar, S. Al-Maskari, X. Li, and C. Pang, "Semi-supervised learning for cyberbullying detection in social networks," in *Australasian Database Conference*. Springer, 2014, pp. 160–171.

[16] B. Nandhini and J. Sheeba, "Cyberbullying detection and classification using information retrieval algorithm," in *Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering & Technology (ICARCSET 2015)*. ACM, 2015, p. 20.

[17] B. S. Nandhini and J. Sheeba, "Online social network bullying detection using intelligence techniques," *Procedia Computer Science*, vol. 45, pp. 485–492, 2015.

[18] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, "Abusive language detection in online user content," in *Proceedings of the 25th international conference on world wide web*. International World Wide Web Conferences Steering Committee, 2016, pp. 145–153.

[19] S. Parime and V. Suri, "Cyberbullying detection and prevention: Data mining and psychological perspective," in *Circuit, Power and Computing Technologies (ICCPCT), 2014 International Conference on*. IEEE, 2014, pp. 1541–1547.

[20] A. G. Prasad, S. Sanjana, S. M. Bhat, and B. Harish, "Sentiment analysis for sarcasm detection on streaming short text data," in *Knowledge Engineering and Applications (ICKEA), 2017 2nd International Conference on*. IEEE, 2017, pp. 1–5.

[21] W. Romsaiyud, K. na Nakornphanom, P. Prasertsilp, P. Nurarak, and P. Konglerd, "Automated cyberbullying detection using clustering appearance patterns," in *Knowledge and Smart Technology (KST), 2017 9th International Conference on*. IEEE, 2017, pp. 242–247.

[22] G. Sarna and M. Bhatia, "Content based approach to find the credibility of user in social networks: an application of cyberbullying," *International Journal Of Machine Learning and Cybernetics*, vol. 8, no. 2, pp. 677–689, 2017.

[23] I.-H. Ting, W. S. Liou, D. Liberona, S.-L. Wang, and G. M. T. Bermudez, "Towards the detection of cyberbullying based on social network mining techniques," in *Behavioral, Economic, Socio-cultural Computing (BESC), 2017 International Conference on*. IEEE, 2017, pp. 1–2.

[24] A. Upadhyay, A. Chaudhari, S. Ghale, S. Pawar *et al.*, "Detection and prevention measures for cyberbullying and online grooming," in *Inventive Systems and Control (ICISC), 2017 International Conference on*. IEEE, 2017, pp. 1–4.

[25] X. Zhang, J. Tong, N. Vishwamitra, E. Whittaker, J. P. Mazer, R. Kowalski, H. Hu, F. Luo, J. Macbeth, and E. Dillon, "Cyberbullying detection with a pronunciation based

convolutional neural network," in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2016, pp. 740–745.

[26] R. Zhao, A. Zhou, and K. Mao, "Automatic detection of cyberbullying on social networks based on bullying features," in *Proceedings of the 17th international conference on distributed computing and networking*. ACM, 2016, p. 43.