SAVITRIBAI PHULE PUNE UNIVERSITY

A PRELIMINARY PROJECT REPORT ON

Detecting Cyberbullying on social media using Machine Learning

SUBMITTED TOWARDS THE
PARTIAL FULFILLMENT OF THE REQUIREMENTS OF

BACHELOR OF ENGINEERING (Computer Engineering)

BY

Student Name:Harsh Agarwal	Exam No:71915137G
Student Name:Jagruti Jadhav	Exam No:72004376L
Student Name:Babita Jaybhaye	Exam No:72004379E
Student Name:Komal Nagar	Exam No:71915163F

Under The Guidance of

Prof. Shrikant Dhamdhere

DEPARTMENT OF COMPUTER ENGINEERING
Parvatibai Genba Moze College of Engineering
Wagholi Pune -412207

Parvatibai Genba Moze College of Engineering
DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the Project Entitled

Detecting Cyberbullying on social media using Machine Learning

Submitted by

Student Name Harsh Agarwal	Exam No: 71915137G
Student Name Jagruti Jadhav	Exam No:72004376L
Student Name Babita Jaybhaye	Exam No:72004379E
Student Name Komal Nagar	Exam No:71915163F

is a bonafide work carried out by Students under the supervision of Prof. Shrikant Dhamdhere and it is submitted towards the partial fulfillment of the requirement of Bachelor of Engineering (Computer Engineering) Project.

Prof. Shrikant Dhamdhere
Internal Guide
Dept. of Computer Engg.

Prof. Shrikant Dhamdhere
H.O.D
Dept. of Computer Engg.

Abstract

With the exponential increase of social media users, cyber bullying has been emerged as a form of bullying through electronic messages. Social networks provide a rich environment for bullies to uses these networks as vulnerable to attacks against victims. Given the consequences of cyber bullying on victims, it is necessary to find suitable actions to detect and prevent it. Recently, deep neural network-based models have shown significant improvement over traditional models in detecting cyberbullying. Also, new and more complex deep learning architectures are being developed which are proving to be useful in various NLP tasks. The model is trained and evaluated on dataset that is provided by Dataturks. The dataset contained 16000 tweets gathered manually annotated by human experts. Selected Twitter-based features namely text and network-based features were used. Several classifiers are trained for determining cyberbullying

Acknowledgments

Please Write here Acknowledgment.Example given as

It gives us great pleasure in presenting the preliminary project report on ***'Detecting Cyberbullying on social media using Machine Learning'****.*

I would like to take this opportunity to thank my internal guide ***Prof. Shrikant Dhamdhere*** *for giving me all the help and guidance I needed. I am really grateful to them for their kind support. Their valuable suggestions were very helpful.*

I am also grateful to ***Prof. Shrikant Dhamdhere****, Head of Computer Engineering Department, Parvatibai Genba Moze College of Engineering, Wagholi Pune-412207 for his indispensable support, suggestions.*

In the end our special thanks to ***Other Person Name*** *for providing various resources such as laboratory with all needed software platforms, continuous Internet connection, for Our Project.*

Harsh Agarwal
Jagruti Jadhav
Babita Jaybhaye
Komal Nagar
(B.E. Computer Engg.)

Keywords

0.1 TECHNICAL KEYWORDS

1. (a) K. Computing methodologies
    i. K.3 ARTIFICAL INTELLIGENCE
        A. k.3.1 Natural Language Processing
        B. Information Extraction
        C. Machine Learning Translation
        D. Discourse, Dialogue and pragmatics
        E. Natural Language Generation
        F. Lexical Semantics
        G. Phonology
        H. Language resources
2. (a) L. Applied Computing
    i. L.5 Law, Social and Behavioral Science
        A. Anthropology
        B. Law
        C. Psychology
        D. Economics
        E. Sociology

INDEX

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

1.1 MOTIVATION OF THE PROJECT

We discuss the motivation in three parts, namely the grave nature of the menace of cyber-bullying, the algorithmic challenges in the fields of machine learning and natural language processing with respect to this problem, as well the dearth of technical solutions to tackle this problem No amount of comfort or time can fully heal the broken hearts of a parent whose child's life has either been tragically ended or has been marred because of cyber-bullying as contributing factor. Any damage done, either mentally or physically or any loss of life due to this phenomenon is frustrating mindless and is a scar upon the face of society at large One of the main motivating factors behind this work is the realization that we as computer scientists can contribute in a meaningful way towards alleviating a very serious social problem, and the dearth of work in the field of computer science in this area affords a unique opportunity to make a influencing contribution

Secondly, the computational detection of cyber-bullying raises unique questions on the many classes of algorithms in the fields of machine learning and natural language processing with respect to the phenomenon of social interaction analysis, especially in the online domain.Motivation of this project is to investigate how one might plug in the opening between these seemingly disparate fields - that an effective parameterization approach to exert the full power and weight of statistical machine learning and natural language processing involves the drawing of relevant parameters from the fields of sociology, psychiatry and sociolinguistics, all three of which have been studying the phenomenon of bullying and unkindness for decades.

Thirdly, we found it both surprising and unsettling to find a complete lack of work in the fields of computational linguistics and human-computer interaction specific to cyber-bullying.

1.2 PROBLEM DEFINITION AND SCOPE

It is important to underline the complexity of the problem of cyber-bullying and carve a crisp problem space that is ripe for the deployment of artificial intelligence and human-computer interaction paradigms. At a fundamental level, bullying amongst

the young is influenced by several social and psychiatric factors. If one were to dig deeper into each such factor, it becomes abundantly clear this is a problem that is rooted in societal norms and cultures.It becomes important to define very clearly what constitutes cyber-bullying.

Cyber-bullying involves a distribution of digital harassment techniques, not limited to but involving the following: uploading of pictures or photos to embarrass a victim, stealing or hacking of personal information such as passwords and user meta information, sending or posting of abusive or damaging messages on social networking websites or through SMS text messages, sexting, making a fake account of an individual on a social network etc. In this project, we limit our work to modeling the detection of textual cyber-bullying: both explicit forms of abuse,implicit or indirect ways of abusing another person, and personal recollections of drama-related anxiety by teenagers.

1.2.1 Scope

True solutions to reduce the problem of cyber-bullying requires a fundamental restructuring of mindsets and cultural change on a huge scale. The purpose of this project is to underline the technology as an ally in mitigating its effects.Teenagers expressing recollections of distressing events can be directed to targeted help or shown messages that might reduce their difficulty.The scope of this project includes finding specific scenarios where an embedding of artificial intelligence can assist help for distressed teenagers, as upon detecting serious cases of cyber-bullying.

1.3 PROBLEM STATEMENT

Cyberbullying is a critical global issue that affects both individual victims and societies. Many attempts have been introduced in the literature to intervene in, prevent, or mitigate cyberbullying; however, because these attempts rely on the victims' interactions, they are not practical. Therefore, detection of cyberbullying without the involvement of the victims is necessary. In this problem we have to classify the statement weather if user is victim of cyberbullying or not

1.3.1 Goals and objectives

Goal and Objectives:

- Implement cyberbullying detection system using given dataset
- To study impact of various standard ml algorithms along with different data processing techniques in improving accuracy

1.3.2 Statement of scope

- Employing machine learning and interaction paradigms to provide an empathetic affordance to users is a research area that currently does not exist in the community. The future of this project is to lay broad-based principles of what that kind of paradigm it involves.

1.3.3 Organization of the report

The whole report is divided into 6 chapters chapters 1 contains the introduction and problem defination while chapter 2 contains of background work and literature review chapter 3 contains requirement analysis and design phase along with prototyping in chapter 4 we have given estimate of project and how scheduling is done the risk analysis of project is also there. In chapter 5 and chapter 6 we have some Uml daigrams to show the design of project.

CHAPTER 2

LITERATURE SURVEY

For several years, the researchers have worked intensively on cyberbully detection to find a way to control or reduce cyberbully in Social Media platforms. Cyberbullying is troubling, as victims cannot cope with the emotional burden of violent, intimidating, degrading, and hostile messages. To reduce its harmful effects, the cyberbullying phenomenon needs to be studied in terms of detection, prevention, and mitigation.

- [1] for instance, reported how through the development of a simple language-specific method, they recorded the percentage of curse and insult words in a post, achieving a recall = 0.785 in cyberbullying identification on a small Formspring dataset.

- [2] developed a program (i.e., BullyTracer) where they identified a "cyberbullying window" 85.3 of the time (recall) and an "innocent window" 51.9 of the time in MySpace posts. More recently, the most common approach to cyberbullying detection has been through feature engineering, which has expanded the common bag-of-words representation of text by creating additional features/dimensions that use domain knowledge of linguistic cues in cyberbullying to attempt to improve a given classical classifier's performance (e.g., Support Vector Machines - SVM, Logistic Regression). Frequent features relate to the use of profanity and how often it occurs in text

- [3] hey used seed words from three categories (abusive, violent, obscene) to calculate SO-PMI IR score and maximized the relevance of categories. Their method achieved 90 of Precision for 10 Recall. We used both of the above methods as a baselines for comparison due to similarities in used datasets and experiment settings. Unfortunately, method by [3], based on Yahoo! search engine API, faced a problem of a sudden drop in Precision

- [4] investigate the performance of several models introduced for cyberbullying detection on Wikipedia, Twitter, and Formspring as well as a new YouTube dataset. They found out that using deep learning methodologies, the performance on YouTube dataset increased

- [5] A recent paper describes similar work is that is being conducted at Massachusetts Institute of Technology. The research is aimed towards detecting cyberbullying through textual context in YouTube video comments. The first level of classification is to determine if the comment is in a range of sensitive topics such as sexuality, race/culture, intelligence, and physical attributes. The second level is determining what topic. The overall success off this experiment was 66.7accuracy for detecting instances of cyberbullying in YouTube comments. This project also used a support vector machine learner

CHAPTER 3

DESIGN DETAILS

3.1 PHASE I : REQUIRED ANALYSIS

Cyber-bullying involves a distribution of digital harassment techniques, not limited to but involving the following: uploading of pictures or photos to embarrass a victim, stealing or hacking of personal information such as passwords and user meta information, sending or posting of abusive or damaging messages on social networking websites or through SMS text messages, sexting, making a fake account of an individual on a social network etc. In this project, we limit our work to model- ing the detection of textual cyber-bullying: both explicit forms of abuse,implicit or indirect ways of abusing another person, and personal recollections of drama-related anxiety by teenagers.so according to problem defination we had three requirements to train system to take the comments file and give result and last one is to take individual comment and predict it.

3.1.1 Format of SRS

Software requirement Specification is a detailed write-up indicating the requirements that the project demands. it contains actual detailed problem definition. The definition is to include all that is to be done and is to be developed in the final software and / or Hardware (product) that is to be generated form the years work (User's point of view). The entries under this section are to be categorized under the categories

- Necessary functions,
- Desirable functions, and others

3.1.2 Requirement

Comment prediction requirement
The system should provide text to feature function which can take the necessary part and obtain a feature vector.
The system should have a well-trained SVM to generate better inputs for classifier.
The system should provide text parser functions which can take the whole text and separate into tokens.
The system needs a classifier which is well trained that predicts the probability of

each sentence.

Web page requirement

The system should provide a button with complete functionality. When clicked on this button, browser send the data from text box to the server.

The function to extract unnecessary data from web and scrap it.

The system should provide communication between server and client with necessary network functions.

Train System Requirements

The system should provide a configuration file for taking new data from admin to train models

Software Interfaces

In this system there will be an API named as CYB API. CYB API is used to preprocess text and for tokenization of text and predicting the outcome of sentence. This is ML API.

Communications Interfaces

The only communication is between the browser and the server. Flask tool will be used to send queries and receive ones. HTTP will be used as the protocol. Hardware interface not available

3.2 PHASE II : ANALYSIS

We had used functional paradigm in this project. This project uses agile model and testing is done after completion of each sprint.

3.3 PHASE IV : PROJECT PLANNING

There would be the requirement of database in the project the database would have overall 12 tables names as experiment-tags, experiments, latest-metrics, metrics, model-version-tags, model-versions, parameters, registered-model-tags, registered-models, runs, tags these table does not required normalization the coding language for selected project would be python, HTML, css and JavaScript for both front end and backend. The agile prototyping model would be used for entire project. The time

estimated for this project is 5 months. Since Line of code for this project 1.5 Kloc. There would be 6 Major functions to be created in backend perspective and these 6 functions are blocks that are mentioned in architecture diagram. These functions can be interlinked with each other. Software and hardware requirement are being mentioned at the end of this chapter. Probable date of completion of this project would be 20 December.The workflow patterns for deployment purpose is the area of software that can be reused

3.4 PHASE V : PROTOTYPING

A GUI interface had two text boxes one for taking suspected comment from the user and the another text box that gives the result of the comment the top of the webpage had tabs and on clicking on cyberbullying button we would get a navigation menu. the following interface after completion would roughly look in 3.1.

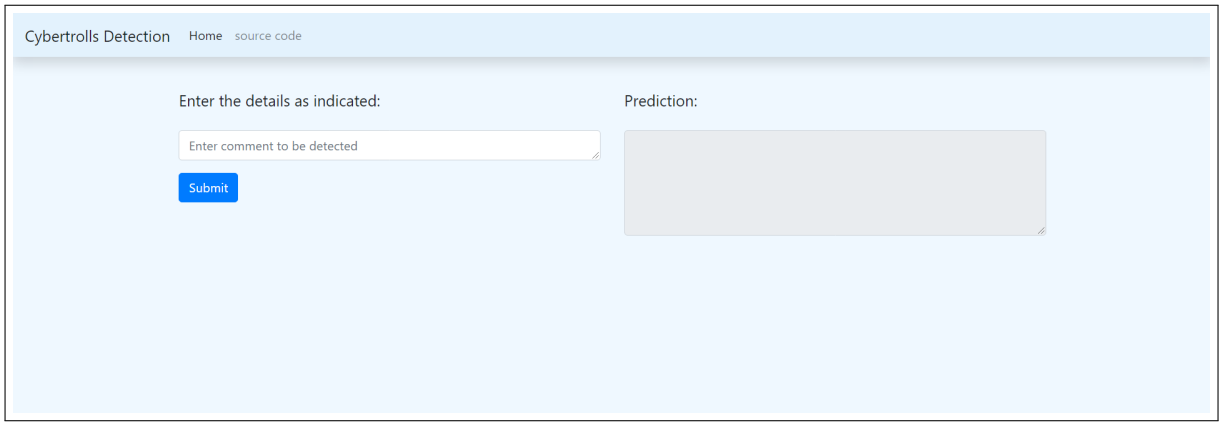

Figure 3.1: Interface diagram

The structure of backend is shown in 6.2 it had 3 main components this components had been defined in form of pipeline in main backend the code had also had done with exception handling, The architecture diagram had shown the prediction service block which is a flask based web app where we used both front end and back end.

3.5 HARDWARE RESOURCES REQUIRED

Below table shows the hardware requirement of the software

Sr. No.	Parameter	Minimum Requirement	Justification
1	CPU Speed	2 GHz	Remark Required
2	RAM	8 GB	Remark Required

Table 3.1: Hardware Requirements
not such

3.6 SOFTWARE RESOURCES REQUIRED

Platform :

1. Operating System: Windows/Linux, 8 GB Ram, 2 Gb hard disk, Gpu
2. IDE: VScode
3. Programming Language: Python, Html, Css, Javascript

CHAPTER 4

CONCLUSION

4.1 PROJECT ESTIMATES

Use agile model and associated streams for estimation.

4.1.1 Reconciled Estimates

4.1.1.1 Cost Estimate

4.1.1.2 Time Estimates

10.4 month

4.1.2 Project Resources

We required total four people which is being flexible in working with two or more roles. along with that we need open source tool i.e. python, GitHub, DVC, mlflow softwares along with that in hardware we required 8GB ram and 4 GB memory is required preferred with graphic card based on Memory Sharing, IPC, and Concurrency.

4.2 RISK MANAGEMENT W.R.T. NP HARD ANALYSIS

This section discusses Project risks and the approach to managing them.

4.2.1 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	End user resist system	Medium	High	Medium	Medium
2	Technology will not meet expectation	Medium	Low	High	Medium
3	Lack of training on tool	High	High	Medium	High
4	Staff inexperienced	Low	Low	Medium	Medium

Table 4.1: Risk Table

Probability	Value	Description
High	Probability of occurrence is	$> 75\%$
Medium	Probability of occurrence is	$26 - 75\%$
Low	Probability of occurrence is	$< 25\%$

Table 4.2: Risk Probability definitions [**?**]

Impact	Value	Description
Very high	$> 10\%$	Schedule impact or Unacceptable quality
High	$5 - 10\%$	Schedule impact or Some parts of the project have low quality
Medium	$< 5\%$	Schedule impact or Barely noticeable degradation in quality Low Impact on schedule or Quality can be incorporated

Table 4.3: Risk Impact definitions [**?**]

4.2.2 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Risk ID	1
Risk Description	Technology does not meet expectation
Category	Development Environment.
Source	Software requirement Specification document.
Probability	Medium
Impact	Medium
Response	The formal meeting must be conducted
Strategy	The team must re-verify the documents and re-plan the requirement
Risk Status	identified

4.3 PROJECT SCHEDULE

This section had detailed project schedule with task set and gantt chart

Risk ID	2
Risk Description	End user resist system
Category	Requirements
Source	Software Design Specification documentation review.
Probability	medium
Impact	Medium
Response	Application should be redeveloped by taking end-user in consideration
Strategy	System must be revaluated and find the reason for failure and take steps according to it
Risk Status	Identified

Risk ID	3
Risk Description	Lack of training on tool
Category	Technology
Source	This was identified during early development.
Probability	High
Impact	High
Response	The development team must be updated with the tools and try to regain experience
Strategy	The team manager must conduct conference to help team
Risk Status	Occured

4.3.1 Project task set

Major Tasks in the Project stages are:

- Task 1: Is to create software environment
- Task 2: Collect data set and create pipeline
- Task 3: Model development and log production
- Task 4: Creating Web application and API development
- Task 5: Starting of test environment and create test cases

Risk ID	4
Risk Description	development team unexperienced
Category	Technology
Source	This was identified during early development.
Probability	Low
Impact	Medium
Response	The development team must be updated with the tools and try to regain experience
Strategy	The experience team must help the weak links
Risk Status	identified

- Task 6: Create workflow and start deployment activity
- task 7: Complete heroku deployment on different branches of repositories

4.3.2 Task network

Project tasks and their dependencies are noted in this diagrammatic format begin-center

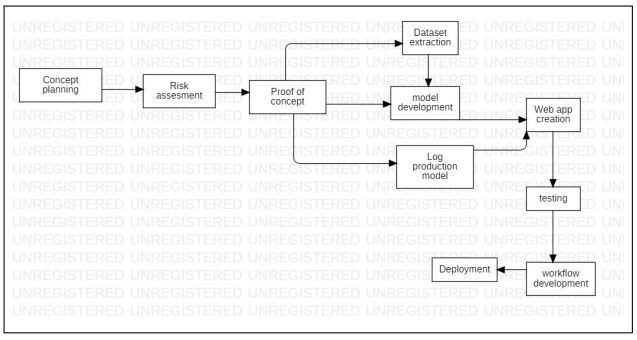

Figure 4.1: Task network

4.3.3 Timeline Chart

A project timeline chart is presented. This may include a time line for the entire project.

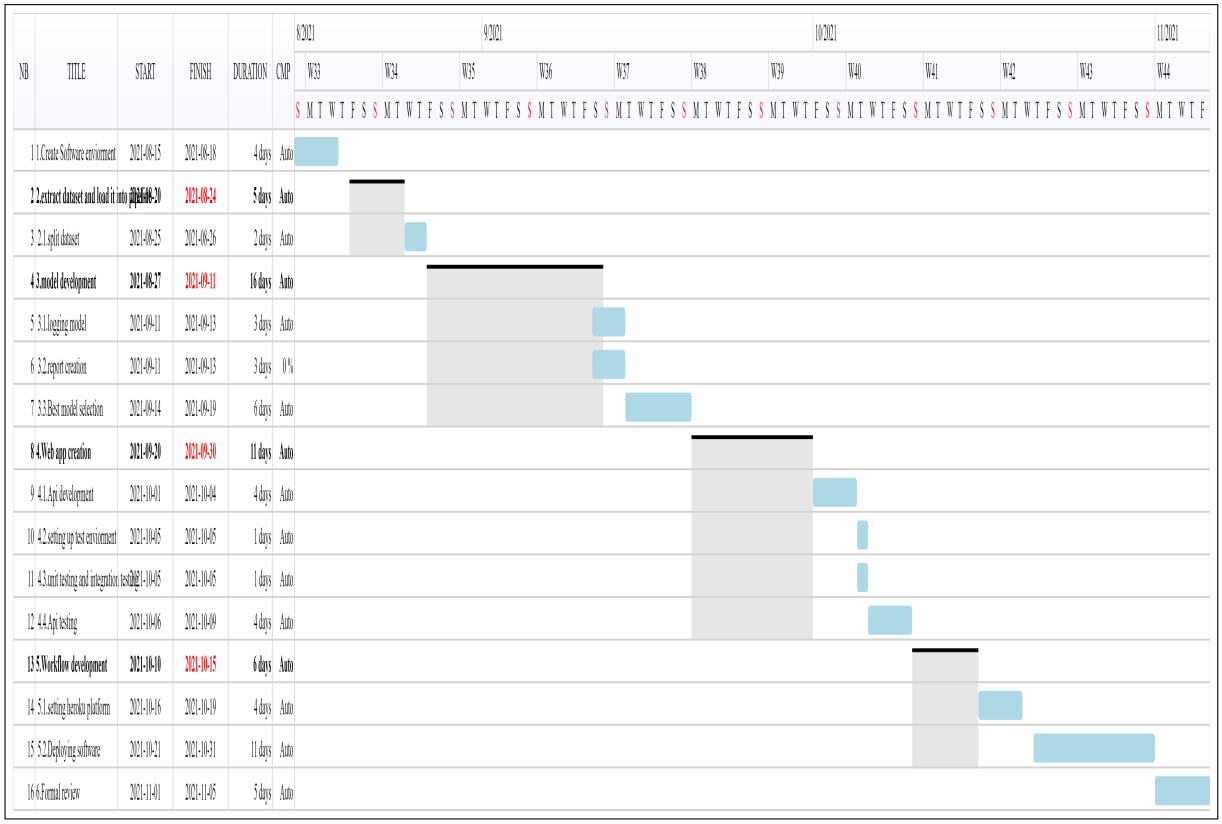

Figure 4.2: Timeline gantt chart

4.4 TEAM ORGANIZATION

The manner in which staff is organized and the mechanisms for reporting are noted.

4.4.1 Team structure

The team structure comprises of various roles the roles given to members include software developer is to create backend of the web application, tester is required for some unit testing and integration testing, UI developer creates the frontend of the project communicated with software developer to create proper communication between them, Nlp engineer is required for creating nlp model and data cleaning, devops enginneer for creating pipelines and deploying it

4.4.2 Management reporting and communication

Mechanisms for progress reporting and inter/intra team communication are identified as per assessment sheet and lab time table.

CHAPTER 5

SAMPLE OF DESIGN DETAILS CHAPTER PHASES - I & II & III SOFTWARE REQUIREMENT SPECIFICATION (SRS IS TO BE PREPARED USING RELEVANT MATHEMATICS DERIVED AND SOFTWARE ENGG.)

5.1 INTRODUCTION

5.1.1 Purpose and Scope of Document

This section of the chapter covers various use cases and functional requirement of the project with help of various UML diagrams

5.1.2 Overview of responsibilities of Developer

What all activities carried out by developer?

5.2 USAGE SCENARIO

5.2.1 User profiles

User: The user sends a request for the text to be checked for cyberbullying. Admin: Admin manages the website and configure a system to send responds to user requests. His/ Her another role is to maintain the algorithm and the server.

5.2.2 Use-cases

Sr No.	Use Case	Description	Actors	Assumptions
1	Webpage	Getting detection of comment	User	User click on button
2	Predict comment	Predict comment from web page	User	Error message will be displayed
3	Train System	Train the model	admin	Admin trains the classifier on new data

Table 5.1: Use Cases

5.2.3 Use Case View

Use Case Diagram. Example is given below

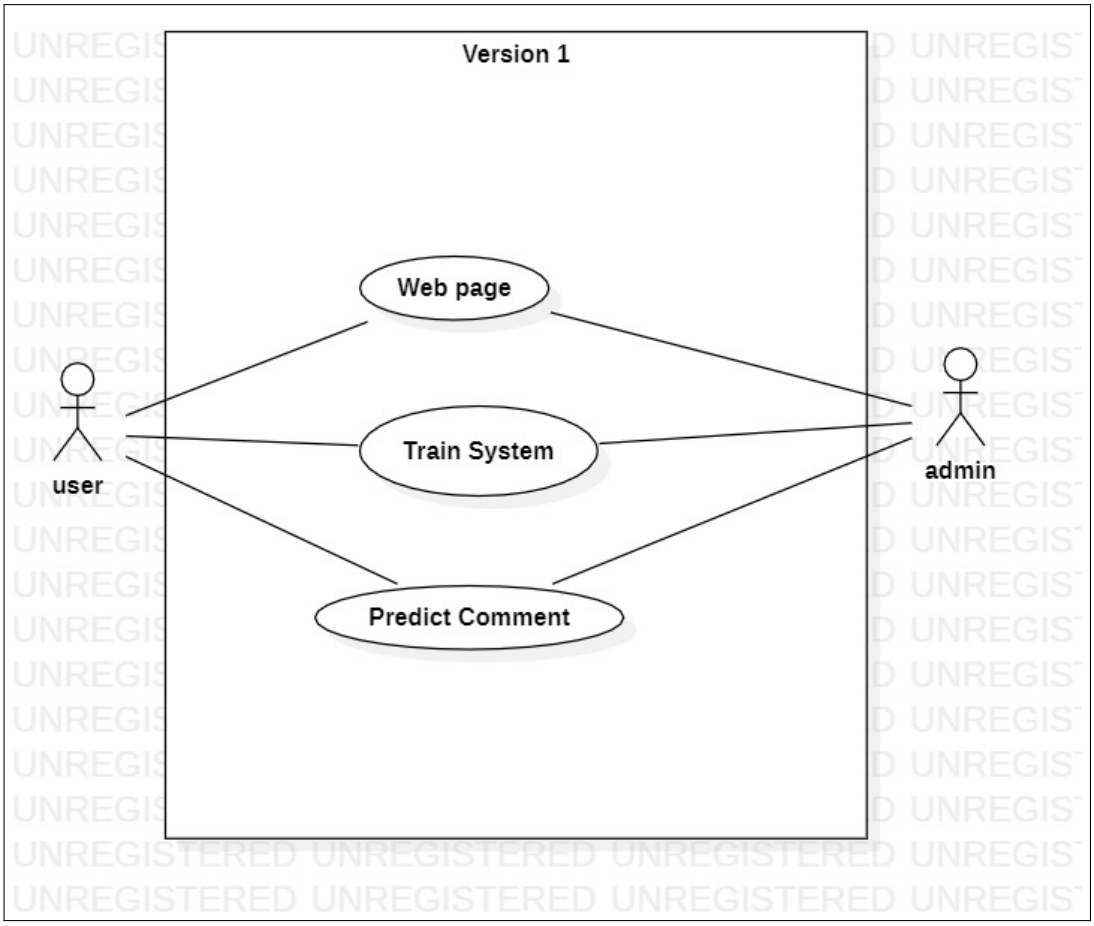

Figure 5.1: Use case diagram

5.3 FUNCTIONAL MODEL AND DESCRIPTION

A description of each major software function, along with data flow (structured analysis) or class hierarchy (Analysis Class diagram with class description for object oriented system) is presented.

5.3.1 Data Flow Diagram

5.3.1.1 Level 0 Data Flow Diagram

5.3.1.2 Level 1 Data Flow Diagram

5.3.2 Description of functions

A description of each software function is presented. A processing narrative for function n is presented.(Steps)/ Activity Diagrams. For Example Refer 5.2

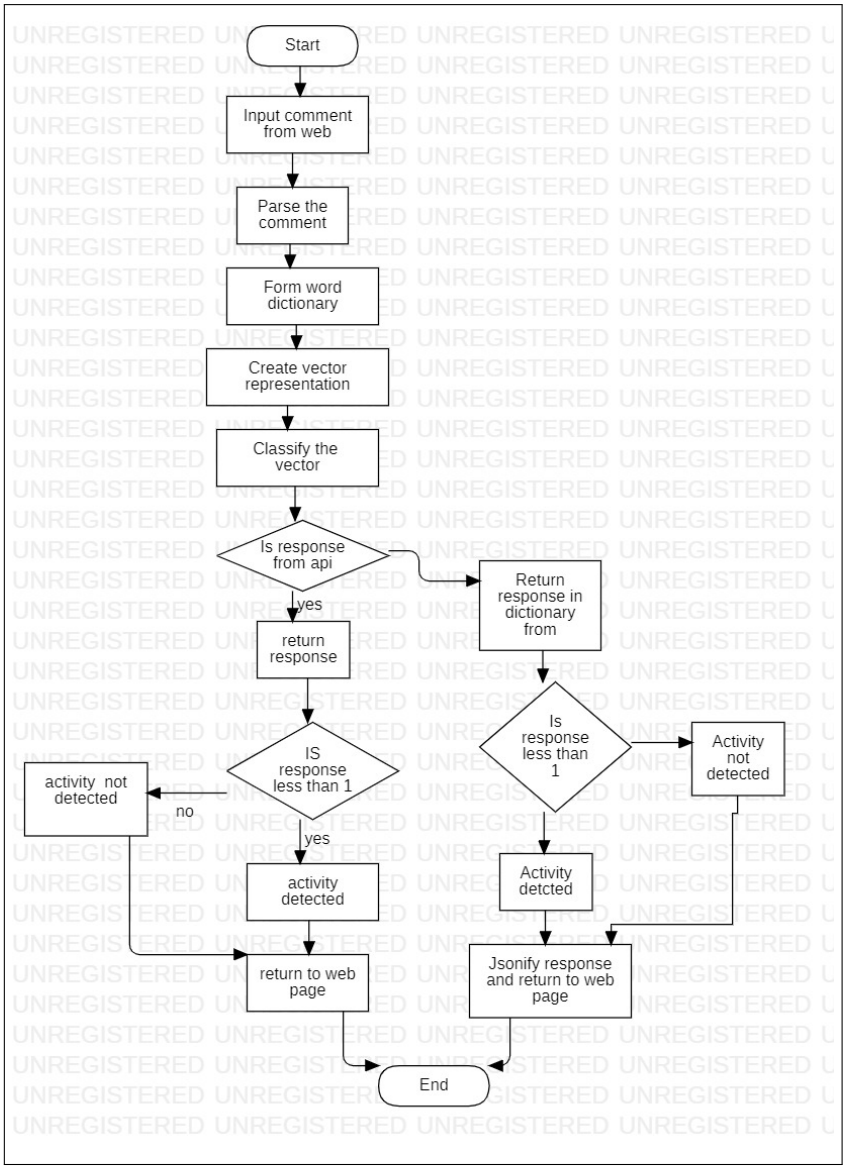

Figure 5.2: flowchart

5.3.3 Activity Diagram:

- The Activity diagram represents the steps taken.

5.3.4 Non Functional Requirements:

Usability:

The system should be easy to use. The user should reach the prediction with one button press if possible. Because one of the software's features is timesaving.

The system also should be user friendly for admins because anyone can be admin instead of programmers.

Training the classifiers is used too many times, so it is better to make it easy.

Reliability:

This software will be developed with machine learning, feature engineering and deep learning techniques. So, in this step there is no certain reliable percentage that is measurable.

Also, user provided data will be used to compare with result and measure reliability. With recent machine learning techniques, user gained data should be enough for reliability if enough data is obtained.

The maintenance period should not be a matter because the reliable version is always run on the server which allow users to access cyberbullying software. When admins want to update, it takes long as upload and update time of executable on server. The users can be reach and use program at any time, so maintenance should not be a big issue.

Performance: Calculation time and response time should be as little as possible, because one of the software's features is timesaving. Whole cycle of detection of comment should not be more than 15 seconds.

The capacity of servers should be as high as possible. Calculation and response times are very low, and this comes with that there can be so many sessions at the same times. The software only used in India, then do not need to consider global sessions.

1 minute degradation of response time should be acceptable. The certain session limit also acceptable at early stages of development. It can be confirmed to user with "servers are not ready at this time" message.

Supportability:

The system should require Python knowledge to maintenance. If any problem acquires in server side and machine learning methods, it requires code knowledge and machine learning background to solve. Client-side problems should be fixed with an update and it also require code knowledge and network knowledge. Software Interfaces In this system there will be an API named as CYB API. CYB API is used to preprocess text and for tokenization of text and predicting the outcome of sentence. This is ML API. there are two api's required one is MLflow which generates

all database and keep track of experiement metrices and dvc to create pipelines and check the track of changes in pipeline and data versioning

5.3.5 Design Constraints

1. Comment should be in English language.
2. OS should support Linux application.
3. User should have web browser to use application
4.All 4 members will work for the project no option for outsource
5.Server shouldn't have any time constraint or should be greater than 10 sec

5.3.6 Software Interface Description

Sofware interface is in form of api and web so we need and web browser along with that we had an api system.

CHAPTER 6

SAMPLE OF CHAPTER DETAILED DESIGN PHASE V

6.1 INTRODUCTION

This document specifies the design that is used to solve the problem of Product.

6.2 ARCHITECTURAL DESIGN

A description of the program architecture is presented. each subsystem is divided into blocks. Data extraction here we just extract data from online database and convert into suitable file format in load data we convert the given file to format that can be easily processed in programming language after that we create another block to split dataset as it is important part of nlp project after that we created various mathematical model with various mathematical model in project to select best mathematical model we create log production model and this model is stored in another folder so that it can be used by prediction service to get the prediction.

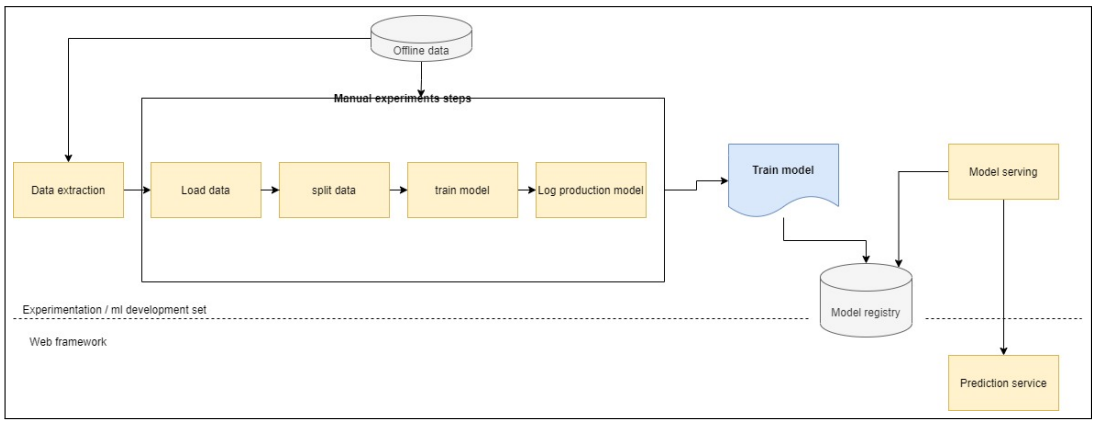

Figure 6.1: Architecture diagram

6.3 DATA DESIGN

A description of all data structures including internal, global, and temporary data structures, database design (tables), file formats.

6.3.1 Internal software data structure

The csv file which is passed through whole manual experiment pipeline till train model and after the pipeline we pass whole artifact while which include model in

deployed format and environment for using in prediction service.

6.3.2 Global data structure

There are two such global data structures that are available to major part of software one that is configuration file which includes project name and addresses where file to be stored and tox.ini file which gives the use of testing environment and build version

6.3.3 Temporary data structure

temporary data structures used in the file report.json and scores,json which had all the parameters used to create model and how the model had performed on test dataset.

6.3.4 Database description

there is one database that is automatically create by ml-flow API which is called ml-flow db it is in sqlite database this contains artifacts since there are various mathematical model so we track the various metrics and conditions from which this model is created along with that we also store result each experiment created has unique experiment number and tags weather it is serving or in staging area this is stored in artifacts stored in separate folder

6.4 COMPONENT DESIGN

Here we had shown component diagram

6.4.1 Component Diagram

Basically whole software can be categorized into three parts as text is received from web we first parse that text and also perform cleaning in text parser component after text is passed through text parser component we get a vector of words
we passed that vector of words to vector creator where we use the frequency dictionary to create a numerical feature vector that can be used by classifier component.

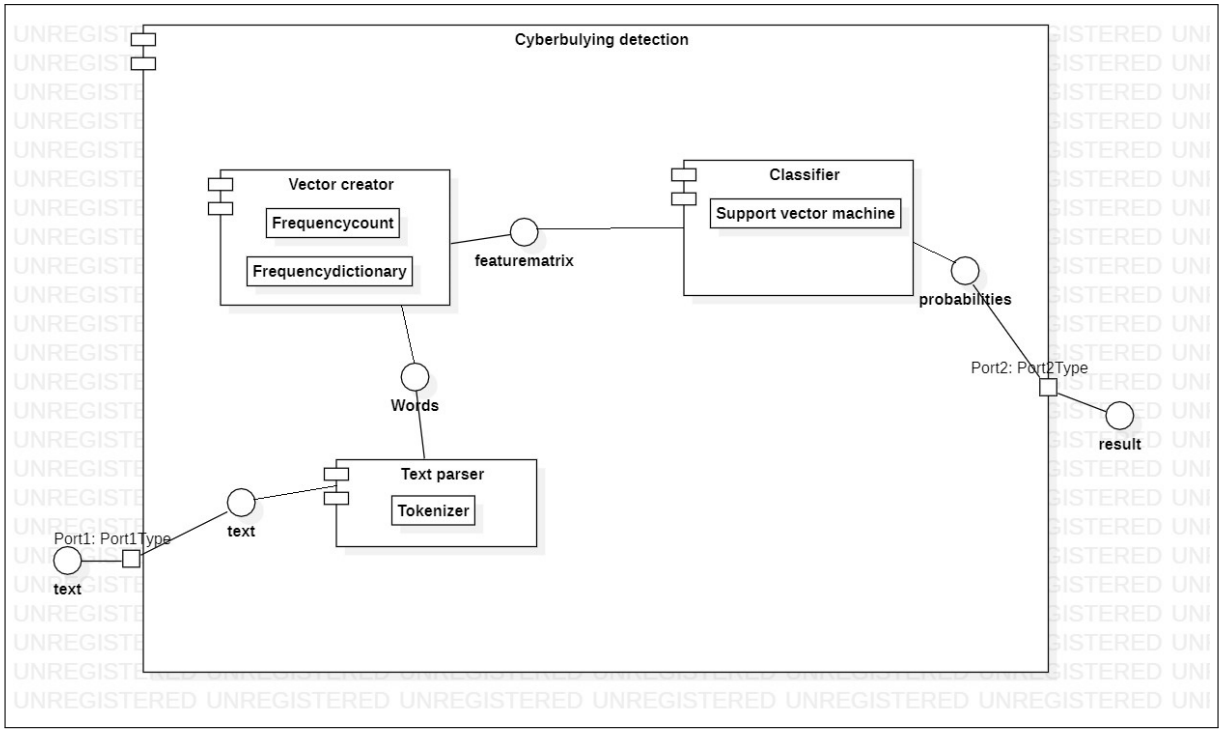

Figure 6.2: Component diagram

Classifier component takes the numerical feature vector and since we had selected SVM model we had get probabilities which is given by decision function of SVM after that depending on probability weather it is greater than 0 or less than 0 we get our result which is given as response to web

CHAPTER 7

REFERENCES

[1] K. Reynolds, A. Kontostathis, and L. Edwards, “Using machine learning to detect cyberbullying,” in *2011 10th International Conference on Machine learning and applications and workshops*, vol. 2, pp. 241–244, IEEE, 2011.

[2] J. Bayzick, A. Kontostathis, and L. Edwards, “Detecting the presence of cyberbullying using computer software,” 2011.

[3] T. Nitta, F. Masui, M. Ptaszynski, Y. Kimura, R. Rzepka, and K. Araki, “Detecting cyberbullying entries on informal school websites based on category relevance maximization,” in *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pp. 579–586, 2013.

[4] R. R. Dalvi, S. B. Chavan, and A. Halbe, “Detecting a twitter cyberbullying using machine learning,” in *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 297–301, IEEE, 2020.

[5] K. Dinakar, R. Reichart, and H. Lieberman, “Modeling the detection of textual cyberbullying,” in *fifth international AAAI conference on weblogs and social media*, 2011.

ANNEXURE A

REVIEWERS COMMENTS OF PAPER SUBMITTED

(At-least one technical paper must be submitted in Term-I on the project design in the conferences/workshops in IITs, Central Universities or UoP Conferences or equivalent International Conferences Sponsored by IEEE/ACM)

1. Paper Title:
2. Name of the Conference/Journal where paper submitted :
3. Paper accepted/rejected :
4. Review comments by reviewer :
5. Corrective actions if any :

ANNEXURE B

PLAGIARISM REPORT

Plagiarism report