**Step1: Introduction:**

## 1.1 Purpose

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

## 1.2 Project Scope

The study will show the effects and the consequence of cyberbullying today in our society, especially to those teen agers who are most affected. This will answer the question about how we can prevent from being cyberbullied. This study will show us how are teens cyberbullied and the fact that being a victim of cyberbullying can be a common and painful experience.

Early detection of harmful social media behaviors such as cyberbullying is necessary for identifying threatening online abnormalities and preventing them from increasing. So, In this project we successfully fetched the comments from the subreddit using praw, and I also able to identify the vulgar comments by using three machine learning algorithm bag of words, term frequency inverse document frequency, support vector machine

**Step 2:**

# Overall Description

## 2.1 Product Perspective

Cyber bullying is the use of technology as a medium to bully someone. Social networking sites provide a fertile medium for bullies, and teens and young adults who use these sites are vulnerable to attack. Through machine learning, we can detect language, patterns used by bullies and their victims and develop rules to automatically detect cyber bullying content.

* Popular social networks like Facebook, Instagram, Twitter and Snapchat
* Text messages sent via direct devices (SMS)
* Instant messaging features offered by e-mail providers, applications or social networks
* Chat rooms and e-mails

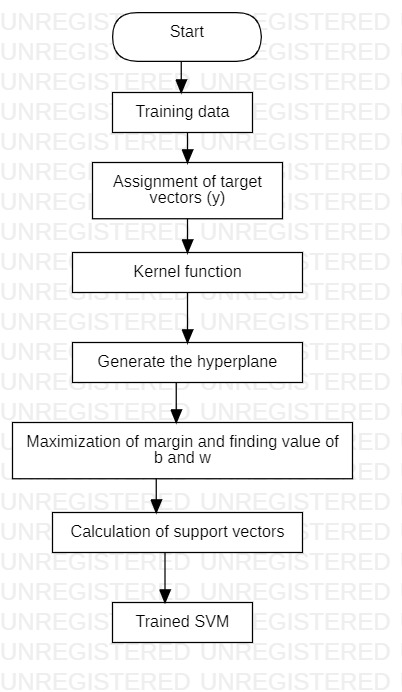
Although symptoms of cyberbullying vary, children and adolescents who are victims of bullying usually have the following symptoms:

* Emotional anger after using the Internet or mobile devices
* Overprotective behavior about digital life
* Getting away from family members, friends and general routine activities
* To exhibit angry and irritated behavior at home

## 

## 2.2 Product Features

Cyberbullying detection is designed using machine learning techniques. Twitter data set is collected with features and labels and mode is trained using **the SVM** algorithm and trained model is applied to live chatting application which has multiple clients and a single server. For each message, cyberbullying is detecting using the model and then alert messages are posted on chat boat.



**2.3 User classes and characteristics:**

* **User**: The user sends a request for the text to be summarized.
* **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.

## 

## 2.4 Operating Environment

Software tools**:**

GitHub Desktop, MlFlow, DVC, Tox, Postman, Heroku

Hardware specification:

Windows 8, 8Gb ram, Intel I5 processor.

## 2.5 Design and Implementation 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

# Step 3:

**3.1** **Use cases and description**

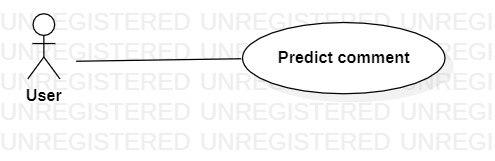
This section includes detailed use cases and characteristics

* **Web page**



|  |  |
| --- | --- |
| Use Case ID | 1 |
| Use Case | Web page |
| Description | Getting detection of comment |
| Actor | User |
| Trigger | User clicks button on web page |
| Primary scenario | User clicks button on web page  System takes the text and send it to server  Comment is parsed and send it as prediction algorithm  Prediction is resent to browser |
| Exceptional Scenario | None |

* **Predict comment**

****

|  |  |
| --- | --- |
| Use Case ID | 2 |
| Use Case | Predict comment |
| Description | Predict comment from web page |
| Actor | User |
| Trigger | User clicks button on web page |
| Primary scenario | User clicks button on web page  System takes the text and send it to server  Comment is parsed and send it as prediction algorithm  Algorithm gives prediction probabilities  Prediction is resent to browser |
| Exceptional Scenario | Error message will be displayed |

* **Train System**



|  |  |
| --- | --- |
| Use Case ID | 3 |
| Use Case | Train system |
| Description | Training the system’s machine learning part for better result |
| Actor | Admin |
| Trigger | Admin enter the system |
| Primary scenario | Admin trains the classifier on new data |
| Exceptional Scenario | None |

**3.2 Functional Requirements of the System:**

Following are the functional requirements on the system:

**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.

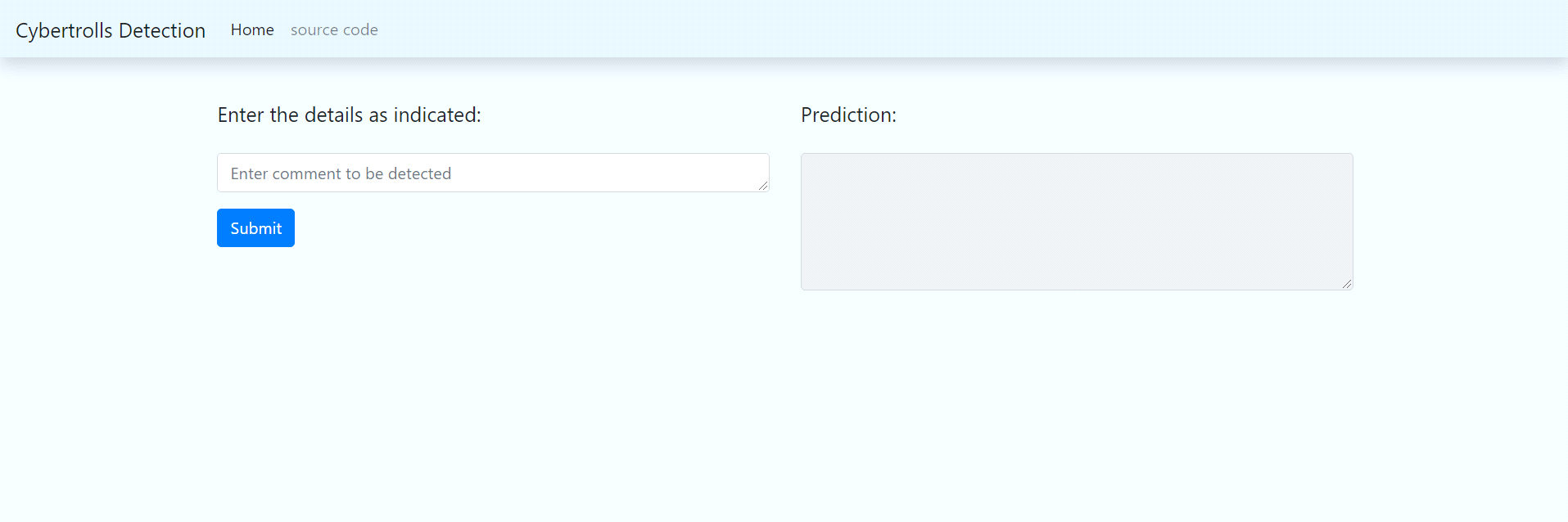
**Train System Requirements**

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

# Step 4: External Interface Requirements

## 4.1 User Interfaces

## User interface had a submit button. When user clicks submit button on a webpage it triggers the prediction function and in the text box it gives the prediction of sentence. The prototype user interface is as follows



**4.2 Hardware Interfaces**

Not applicable

**4.3 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.

## 4.4 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.

**Step 5 Nonfunctional Requirements**

* 1. **Usability:**

1. 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.

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

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

* 1. **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.

* 1. **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.

**5.4 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.