# DESIGN AND ANALYSE THE FRAMEWORK FOR PREVENTING CYBERBULLING IN SOCIAL NETWORKING SITES USING DEEP LEARNING MODEL

[1]Dinesh C , [2]Kalaivani S

1Department of Computer Application, Hindusthan College of arts and science, Coimbatore, Tamilnadu, India.

2Assistant Professor, Department of Computer Application, Hindusthan College of arts and science, Coimbatore,

## ABSTRACT

Cyberbullying is bullying that takes place over digital devices like cell phones, computers, and tablets. Cyberbullying can occur through SMS, Text, and apps, or online in social media, forums, or gaming where people can view, participate in, or share content. Cyberbullying includes sending, posting, or sharing negative, harmful, false, or mean content about someone else. It can include sharing personal or private information about someone else causing embarrassment or humiliation. The content an individual share online – both their personal content as well as any negative, mean, or hurtful content – creates a kind of permanent public record of their views, activities, and behaviour. To avoid or detecting cyberbullying attacks, many existing approaches in the literature incorporate Machine Learning and Natural Language Processing text classification models without considering the sentence semantics. The main goal of this project is to overcome that issue. This project proposed a model LSTM - CNN architecture for detecting cyberbullying attacks and it used word2vec to train the custom of word embeddings. This model is used to classify tweets or comments as bullying or non-bullying based on the toxicity score. LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series. A convolutional neural network (CNN) is a type of artificial neural network and it has a convolutional layer to extract information by a larger piece of text and by using this model LSTM- CNN achieve a higher accuracy in analysis, classification and detecting the cyberbullying attacks on posts and comments.

# CHAPTER 1

# INTRODUCTION

## 1.1. Overview of the project

Cyberbullying is bullying that takes place over digital devices like cell phones, computers, and tablets. Cyberbullying can occur through SMS, Text, and apps, or online in social media, forums, or gaming where people can view, participate in, or share content. Cyberbullying includes sending, posting, or sharing negative, harmful, false, or mean content about someone else. It can include sharing personal or private information about someone else causing embarrassment or humiliation. Some cyberbullying crosses the line into unlawful or criminal behaviour.

## 1.2 MODULE DESCRIPTION:

## 1 Social Networking Web App

Build a social networking service is an online platform which people use to build social networks or social relationships with other people who share similar personal or career interests, activities, backgrounds or real-life connections. Social networking services vary in format and the number of features. The classification model has been exposed as a REST API which was consumed by a Web application built using Python's Flask framework. The main features include an Admin dashboard for visualization of cyberbullying activities, an option to search tweets, and automatic generation and emailing of reports of cyberbullying activity.

### 2. Aadhar User Account Management

- **New User**

Create user account with aadhar number.

- **Existing User**

The existing users of Facebook will also have to upload a scanned copy of their Aadhar Card. If they fail to do so, their profile will be suspended within the next 15 days.

### 2.1 Cyberbullying Analysis API

In this module we developed the API for cyberbullying analytics on chat or post user data. It focuses on keywords and analyzes chat or post according to a two-pole scale (positive and negative).

## 3. Training Phase: Cyberbullying Tweet Classification

### 3.1 Cyberbullying Data Set Annotation

We used cyberbullying data from Kaggle. The dataset in consisted of two labels, positive and negative, while was composed of three labels of positive, neutral, and negative. Furthermore, the dataset in was composed of five labels of positive, somewhat positive, neutral, somewhat negative, and negative.

### 3.2 Pre-processing

Datasets contain unnecessary data in raw form that can be unstructured or semi-structured. Such unnecessary data increases training time of the model and might degrades its performance. Pre-processing plays a vital role in improving the efficiency of DL models and saving computational resources. Text pre-processing boosts

the prediction accuracy of the model. The preprocessing step is essential in cyberbullying detection. It consists of both cleaning of texts (e.g., removal of stop words and punctuation marks), as well as spam content removal. In the proposed model, it has been applied to remove and clean unwanted noise in text detection. For example, stop words, special characters, and repeated words were removed. Then, the stemming for the remaining words to their original roots has been applied as a result of this preprocessing, and the dataset containing clean tweets is produced for the proposed model to be run and predicted. In this project, we executed various data preprocessing steps such as tokenization, spelling correction, stop words removing, punctuation removing, digit removing, removing a non-Bangla character, removing Emoticons, word normalization, and lemmatization and data splitting.



**Tokenization:** Tokenization refers to splitting up a sentence, phrase, or word into numerous smaller linguistic units named "Tokens". These tokens help to comprehend the NLP model and decipher the significance of the content by investigating the arrangement of the words. In natural language processing, the language needs to be analyzed and scrutinized under certain constraints and conditions. Tokenization breaks up a text to facilitate the whole process of analyzing a language in detail.

**Spelling correction:** The collected text data was full of spelling blunders, and hence the raw data had to undergo a decent scrutinization. In this dataset, most of the errors were type-error. Consequently, we made corrections to these words. This process was primarily done by a deep learning-based Bengali spell checker module which used the "Bangla Academy" word database as the training system. We pass every article and get a text without spelling errors.

**Stop words removal**: Stop words are the words being used in a language without coordinating meaningful information. In English, there are a handful of stop words that do not denote any particular meaning rather than helping another phrase or words to make sense of. In order to train a model, it's very important to remove the stop words since stop words exist in a high quantity without providing any meaningful and unique sense.

**Punctuation removal:** Punctuation can play an important role when it comes to creating an emotional vibe to the expression. Apart from that, punctuation has no role when data has to be converted to a word embedding method. In our project, the word embedding method was incorporated, and for that reason, punctuations were to be removed beforehand.

**Digit removing:** The data used in this investigation was mostly based on news coverage or bulletins in written form. There existed plenty of numerical characters inscribed in many forms, such as the date and time of an event, address, phone number, and game score, and so on. Generally, these numerical values do not provide any contextual information that can be used to classify the contents. Therefore, all sorts of numerical characters were discarded before training by the models.

**Non-English characters removal**: The core purpose of our research was grounded on using an English corpus. Nevertheless, there existed a lot of Non-English Characters. They were insignificant to the objective of our project and negligible for the scheme we have propounded. Therefore, if there were any Non-English Characters, we handled them by either translating them into relevant Bengali characters or by discarding them depending on the context and intendment of those characters.

**Emoticon removal:** The Internet has opened the horizon of globalization through social media. Emoticons play a vital role in expressing situational emotion while communicating with another person on social media. Nonetheless, the emoticons do not convey a message and do not contain any meaning themselves. Thereupon, Emoticons were removed from the corpus for further investigation.

**Word normalization**: Many words were not used in their standard form in our corpus. Some of the spellings were distorted, and some of them were in an informal format. For example, Bengali currency Taka is sometimes written in "tk". We converted those words into their original spelling automatically by python programming. This process is named word normalization, and it is a prerequisite for a well-developed NLP model.

**Lemmatization**: Lemmatization is the method of changing a word to its base structure. The contrast between stemming and lemmatization will be, lemmatization considers the unique circumstance and converts the word to its significant base structure, while stemming simply eliminates the last couple of characters, frequently prompting off base implications and spelling blunders. Lemmatization in phonetics is the way toward gathering the versatile types

of a word so they can be broken down as a solitary thing, recognized by the word's lemma or word reference structure.

**Data splitting**: Training, validation, and testing are supposed to be the most important phase in the realm of machine learning. In order to train the models, the corpus was methodized in a way, so it becomes compatible with the computation process of the algorithms. Training allows the models to learn from their respective trials and error. Our dataset was split into three segments such as 70% and 10%, and 20%. 70% of our dataset was used for training the models, and the remaining 10% was kept for validation purposes, and the rest 20% is for the testing model. Validation helps the model to evaluate itself and grease the wheels for training repeatedly. After the models were trained, the testing data was used as the testing set for testing the model's performance.

## 2.SYSTEM SPECIFICATION

## 2.1 HARDWARE SPECIFICATION

- Processors: Intel® Core™ i5 processor 4300M at 2.60 GHz or 2.59 GHz (1 socket, 2 cores, 2 threads per core), 8 GB of DRAM
- Disk space: 320 GB
- Operating systems: Windows® 10, macOS*, and Linux*

## 2.2 SOFTWARE SPECIFICATION

- Server Side      : Python 3.7.4(64-bit) or (32-bit)
- Client Side      : HTML, CSS, Bootstrap
- IDE              : Flask 1.1.1
- Back end         : MySQL 5.
- Server           : Wampserver 2i
- OS               : Windows 10 64 –bit or Ubuntu 18.04 LTS "Bionic Beaver"

## 2.3 ABOUT THE SOFTWARE

### Overview of the Frond end

**PYTHON**

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). This tutorial gives enough understanding on Python programming language.



Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages. Python is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain.

Python is currently the most widely used multi-purpose, high-level programming language. Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java. Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time. Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc. The biggest strength of Python is huge collection of standard libraries which can be used for the following:

- Machine Learning
- GUI Applications (like Kivy, Tkinter, PyQt etc.)
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)
- Image processing (like OpenCV, Pillow)
- Web scraping (like Scrapy, Beautiful Soup, Selenium)

- Test frameworks
- Multimedia
- Scientific computing
- Text processing and many more.

**Pandas**

Pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language. pandas is a Python package that provides fast, flexible, and expressive data structures designed to make working with "relational" or "labeled" data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real world data analysis in Python.
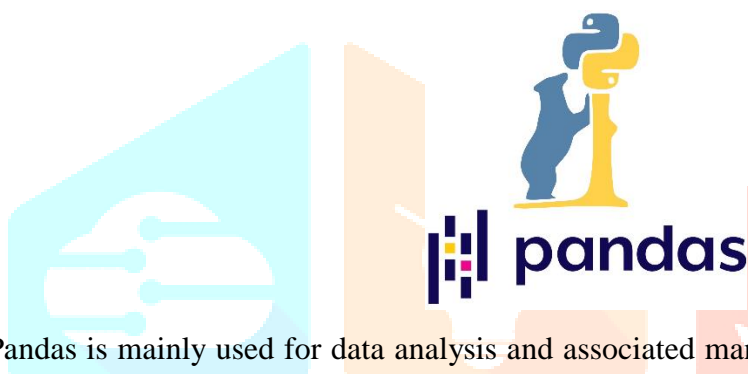


Pandas is mainly used for data analysis and associated manipulation of tabular data in Data frames. Pandas allows importing data from various file formats such as comma-separated values, JSON, Parquet, SQL database tables or queries, and Microsoft Excel. Pandas allows various data manipulation operations such as merging, reshaping, selecting, as well as data cleaning, and data wrangling features. The development of pandas introduced into Python many comparable features of working with Data frames that were established in the R programming language. The panda's library is built upon another library NumPy, which is oriented to efficiently working with arrays instead of the features of working on Data frames.

**NumPy**

NumPy, which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed.



NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

**Matplotlib**

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.

**Seaborn**

Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures. Visualization is the central part of Seaborn which helps in exploration and understanding of data.



Seaborn offers the following functionalities:

- Dataset oriented API to determine the relationship between variables.
- Automatic estimation and plotting of linear regression plots.
- It supports high-level abstractions for multi-plot grids.
- Visualizing univariate and bivariate distribution.

**Scikit Learn**

scikit-learn is a Python module for machine learning built on top of SciPy and is distributed under the 3-Clause BSD license.



Scikit-learn (formerly scikits. learn and also known as sklearn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support-vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

**NLTK**

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum.

NLTK (Natural Language Toolkit) Library is a suite that contains libraries and programs for statistical language processing. It is one of the most powerful NLP libraries, which contains packages to make machines understand human language and reply to it with an appropriate response.

**WordCloud**

A word cloud (also called tag cloud or weighted list) is a visual representation of text data. Words are usually single words, and the importance of each is shown with font size or color. Python fortunately has a wordcloud library allowing to build them.

The wordcloud library is here to help you build a wordcloud in minutes. A word cloud is a data visualization technique that shows the most used words in large font and the least used words in small font. It helps to get an idea about your text data, especially when working on problems based on natural language processing.

**Over view of the Back end**

**MySQL**

MySQL tutorial provides basic and advanced concepts of MySQL. Our MySQL tutorial is designed for beginners and professionals. MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software under the GNU license. It is supported by Oracle Company. MySQL database that provides for how to manage database and to manipulate data with the help of various SQL queries. These queries are: insert records, update records, delete records, select records, create tables, drop tables, etc. There are also given MySQL interview questions to help you better understand the MySQL database.

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications. It is developed, marketed, and supported by MySQL AB, a Swedish company, and written in C programming language and C++ programming language. The official pronunciation of MySQL is not the My Sequel; it is My Ess Que Ell. However, you can pronounce it in your way. Many small

and big companies use MySQL. MySQL supports many Operating Systems like Windows, Linux, MacOS, etc. with C, C++, and Java languages.

## WampServer

WampServer is a Windows web development environment. It allows you to create web applications with Apache2, PHP and a MySQL database. Alongside, PhpMyAdmin allows you to manage easily your database.



WampServer is a reliable web development software program that lets you create web apps with MYSQL database and PHP Apache2. With an intuitive interface, the application features numerous functionalities and makes it the preferred choice of developers from around the world. The software is free to use and doesn't require a payment or subscription.

## Bootstrap 4

Bootstrap is a free and open-source tool collection for creating responsive websites and web applications. It is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites.



It solves many problems which we had once, one of which is the cross-browser compatibility issue. Nowadays, the websites are perfect for all the browsers (IE, Firefox, and Chrome) and for all sizes of screens (Desktop, Tablets, Phablets, and Phones). All thanks to Bootstrap developers -Mark Otto and Jacob Thornton of Twitter, though it was later declared to be an open-source project.

**Easy to use**: Anybody with just basic knowledge of HTML and CSS can start using Bootstrap

**Responsive features**: Bootstrap's responsive CSS adjusts to phones, tablets, and desktops

**Mobile-first approach**: In Bootstrap, mobile-first styles are part of the core framework

**Browser compatibility**: Bootstrap 4 is compatible with all modern browsers (Chrome, Firefox, Internet Explorer 10+, Edge, Safari, and Opera)

**Flask**

Flask is a web framework. This means flask provides you with tools, libraries and technologies that allow you to build a web application. This web application can be some web pages, a blog, a wiki or go as big as a web-based calendar application or a commercial website.



Flask is often referred to as a micro framework. It aims to keep the core of an application simple yet extensible. Flask does not have built-in abstraction layer for database handling, nor does it have formed a validation support. Instead, Flask supports the extensions to add such functionality to the application. Although Flask is rather young compared to most Python frameworks, it holds a great promise and has already gained popularity among Python web developers. Let's take a closer look into Flask, so-called "micro" framework for Python.

Flask was designed to be easy to use and extend. The idea behind Flask is to build a solid foundation for web applications of different complexity. From then on you are free to plug in any extensions you think you need. Also you are free to build your own modules. Flask is great for all kinds of projects. It's especially good for prototyping.

Flask is part of the categories of the micro-framework. Micro-framework are normally framework with little to no dependencies to external libraries. This has pros and cons. Pros would be that the framework is light, there are little dependency to update and watch for security bugs, cons is that some time you will have to do more work by yourself or increase yourself the list of dependencies by adding plugins. In the case of Flask, its dependencies are:

**WSGI-**Web Server Gateway Interface (WSGI) has been adopted as a standard for Python web application development. WSGI is a specification for a universal interface between the web server and the web applications.

**Werkzeug-**It is a WSGI toolkit, which implements requests, response objects, and other utility functions. This enables building a web framework on top of it. The Flask framework uses Werkzeug as one of its bases.

**Jinja2** Jinja2 is a popular templating engine for Python. A web templating system combines a template with a certain data source to render dynamic web pages.


## 3. SYSTEM STUDY

## 3.1. Existing System

In this chapter existing machine learning classifiers utilized for tweet classification will be discussed. This chapter analysed five supervised machine learning algorithms: Support Vector Machines (SVM), Naive Bayes (NB), Random Forest (RF), Decision Tree (DT), Gradient Boosting model (GBM), Logistic Regression (LR) and Voting Classifier (Logistic Regression C Stochastic Gradient Descent classifier).

## Disadvantages

- Process of reporting such cases is long, tedious job.

- Difficult to track.

- Most of the cyberbullying cases go unreported.

- Low accuracy.

- Time consuming process.

## 3.2. Proposed System

In this paper, we design a model based on the bidirectional BiLSTM to detect cyberbullying in textual form.

## Advantages

- It successfully classifies the tweets in various classes.

- Auto report generator generates a simple report for probable accusers.

- Several analytics and report can be sent to the crime department.

- Accuracy is high.

- Foul language on any given page, removes it, and can highlight words as well,
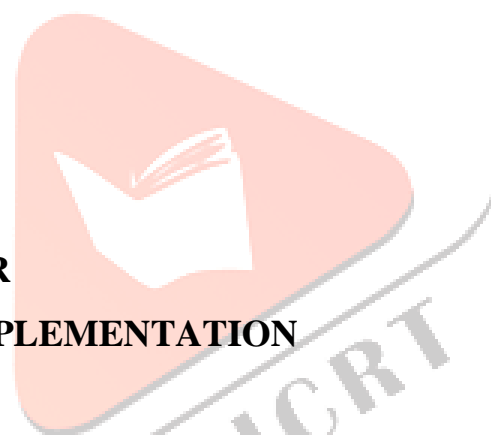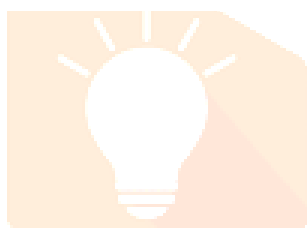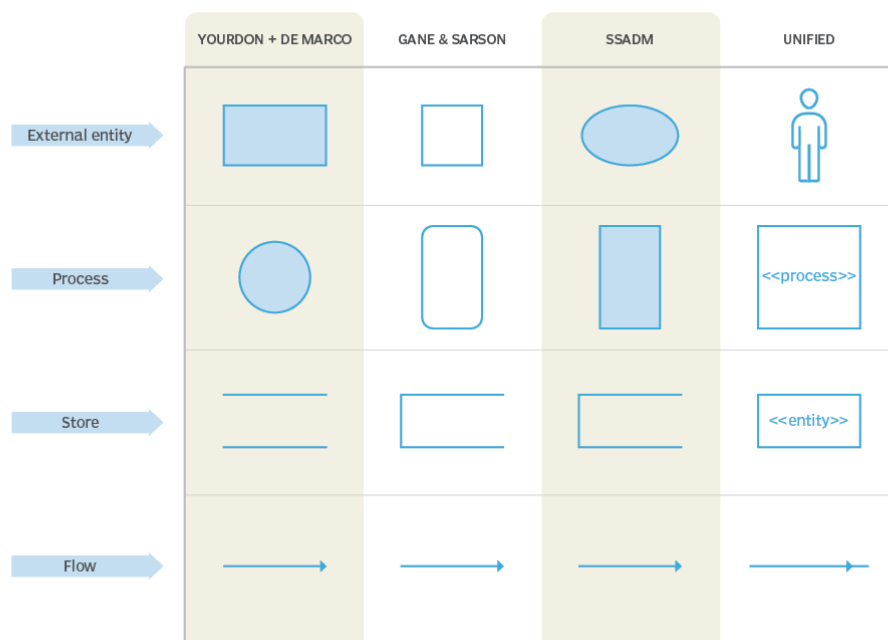
## 4. SYSTEM DESIGN

## 4.1 Database cyber_bullying

## Table structure for table admin

| Field | Type | Null | Default |
|-------|------|------|---------|
| username | varchar(25) | Yes | NULL |
| password | varchar(50) | Yes | NULL |

## 4.2 Data Flow Symbols

Data Flow Diagram symbols are standardized notations, like **rectangles, circles, arrows, and short-text labels**, that describe a system or process' data flow direction, data inputs, data outputs, data storage points, and its various sub-processes.

| | YOURDON + DE MARCO | GANE & SARSON | SSADM | UNIFIED |
|---|---|---|---|---|
| External entity | ▭ | ▭ | ⬭ | 🧍 |
| Process | ● | ▢ | ▮ | <<process>> |
| Store | ▭ | ▭ | ▭ | <<entity>> |
| Flow | → | → | → | → |

# 5 CHAPTER
# SYSTEM TESTING AND IMPLEMENTATION

## 5.1 Testing

In this phase of methodology, testing was carried out on the several application modules. Different kind of testing was done on the modules which are described in the following sections. Generally, tests were done against functional and non-functional requirements of the application following the test cases. Testing the application again and again helped it to become a reliable and stable system.

### 5.1.1 Usability Testing

This was done to determine the usability of the application that was developed. This helped to check whether the application would be easy to use or what pitfalls would the users come through. This was used to determine whether the application is user friendly. It was used to ascertain whether a new user can easily understand the application even before interacting with it so much. The major things checked were: the system flow from one page to another, whether the entry points, icons and words used were functional, visible and easily understood by user.

### 5.1.2 Functional Testing

Functional Testing is defined as a type of testing which verifies that each function of the software application operates in conformance with the requirement specification. This testing mainly involves black box testing and it is not concerned about the source code of the application. Functional tests were done based on different kind of features and modules of the application and observed that whether the features are met actual project objectives and the modules are hundred percent functional. Functional tests, as shown in the following Table-

1 to Table-5, were done based on use cases to determine success or failure of the system implementation and design. For each use case, testing measures were set with results being considered successful or unsuccessful. Below are the tables which are showing some of the major test cases along with their respective test results.

### Table 5.1: Signup/Registration Test Case

| Identifier | Test Case-1 |
|---|---|
| Test Case | Signup |
| Description | To register new account in the application. |
| Pre-requisite | 1) Username and email must not exist previously. |
| Test procedure | 1) Select Sign Up from the menu. Fill in username, email, and password and retype password accordingly. 3) Click on Sign Up button |
| Expected Result | 1) User can register to the application successfully. Username, email and password stored in the user table in the database. |
| Pass/Fail | Pass |

### Table 5.2: Login Test Case

| Identifier | Test Case-2 |
|---|---|
| Test Case | Login |
| Description | To login new account in the application |
| Pre-requisite | 1) Registration must be done previously. |
| Test procedure | 1) Select Log In from the menu. 2) Fill in username and password accordingly. 3) Click on Log In button. |
| Expected Result | 1) User can login to the application successfully. 2) User should access the application features which are allowed |
| Pass/Fail | Pass |

# 6 .CONCLUSION

## 6.1 Conclusion

Cyberbullying is the harassment that takes place in digital devices such as mobile phones, computers and tablets. The means used to harass victims are very diverse: text messages, applications, social media, forums or interactive games. One of the things that complicates these types of situations that occur through the Internet, is the anonymity this environment allows. Since this facilitates cyberbullying can cover almost all areas of the victim's life, that is: educational environment, work, social or loving life. When the identity of the harasser is not known, even if the facts are reported, in many cases it is not enough to open an investigation, identify it and pay for the crime committed. This project proposed a deep learning model Bidirectional Long Short Term Memory (BiLSTM. Thus, this project has designed a method of automatically detecting the Cyberbullying attack cases. Identifies the messages or comments or posts which the BiLSTM model predicts as offensive or negative then it blocks that person id, then the admin can create automated reports and send to the concern department. Experiments are conducted to test three machine learning and 2 deep learning models that are; (1) GBM, (2) LR, (3) NB, (4) LSTM-CNN and (5) BiLSTM. This project also employed two feature representation techniques Tf and TF-IDF. The results showed that all models performed well on tweet dataset but our proposed BiLSTM classifier outperforms by using both TF and TF-IDF among all. Proposed model achieves the highest results using TF-IDF with 96% Accuracy, 92% Recall and 95% F1-score.