A STUDY OF HATE SPEECH DETECTION IN ONLINE FORUMS USING NLP

BY

FAIZA NOSHIN TITHI
ID: 203-15-3877
AND

RITWIKA DEY RISHA
ID: 203-15-3857

FINAL YEAR DESIGN PROJECT REPORT

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Abdus Sattar
Assistant Professor
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised By

Md. Atik Asif Khan Akash
Lecturer
Department of Computer Science and Engineering
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

July 2024

APPROVAL

This Project titled "A study of hate speech detection in online forums using NLP ", submitted by **Faiza Noshin Tithi** and **Ritwika Dey Risha** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **July 15, 2024.**

BOARD OF EXAMINERS

Narayan Ranjan Chakraborty — **Board Chairman**
Associate Professor & Associate Head
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Mushfiqur Rahman — **Internal Examiner 1**
Sr. Lecturer
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

15.7.24

Israt Jahan — **Internal Examiner 2**
Sr. Lecturer
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

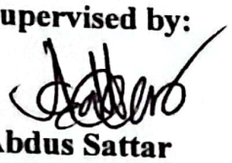
15.07.24

Dr. Abu Sayed Md. Mostafizur Rahaman — **External Examiner**
Professor
Department of Computer Science and Engineering
Jahangirnagar University

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Abdus Sattar, Assistant Professor, Department of Computer Science and Engineering, Daffodil International University**. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:



Abdus Sattar
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised by:

Md. Atik Asif Khan Akash
Lecturer
Department of CSE
Daffodil International University

Submitted by:

Faiza Noshin Tithi
ID: 203-15-3877
Department of CSE
Daffodil International University

Ritwika Dey Risha
ID: 203-15-3857
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to almighty for His divine blessing making it possible for us to complete the final year project/internship successfully.

We are grateful and wish our profound indebtedness to **Abdus Sattar**, **Assistant Professor**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of 'Networking and Security' to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to the **Head of the Department of CSE**, for his kind help in finishing our project and also to other faculty members and the staff of the Department of CSE, Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discussion while completing the course work.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

Hate speech detection has emerged as a critical topic of research in online platforms, to mitigate the negative consequences of discriminatory language and promote a safer digital environment. Using advances in Natural Language Processing (NLP), academics developed a variety of ways for automatically recognizing and categorizing hate speech in text data. This paper provides a detailed assessment of hate speech detection systems based on AI methods, highlighting significant methodologies, problems, and achievements in the field. We will use CNN, RNN and LSTM models to get the best accuracy.Each of these models has its unique strengths and is suited for different types of tasks within the field of deep learning.We begin by looking at the fundamental methodology used in hate speech identification, such as feature engineering, supervised machine learning algorithms, and language analysis tools. Feature engineering is critical for collecting the semantic and historical context required for recognizing hate speech, whereas supervised machine learning algorithms enable model training to distinguish between hate speech vs free speech instances. Furthermore, linguistic analysis techniques such as sentiment analysis and syntactic parsing help to extract significant aspects from text data.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

NLP	Natural Language Processing
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory

CHAPTER 1
INTRODUCTION

1.1 Overview

The development of social networking sites as well as online communities in the age of technology have altered conversation and permitted worldwide discussion and exchanging of ideas. Nevertheless, discriminatory language has proliferated because of that that was unparalleled belonging, creating difficulties for leaders of platforms and virtual communities. As an outcome, studies regarding hate speech registration on online discussion boards have grown critical, automated systems that can recognize and remove such information instantly.

1.2 Background and Present State

Hate speech threatens online harmony and inclusivity. Hate speech, which demeans individuals or groups based on attributes such as race, religion, ethnicity, gender, sexual orientation, or disability, has found fertile ground in the anonymity of online forums. To combat this, researchers have turned to Natural Language Processing (NLP) for automated hate speech detection. NLP leverages linguistic features, machine learning algorithms, and sentiment analysis to identify hateful content. Current systems employ methods such as supervised learning, which uses labeled datasets to train models, and deep learning, which captures complex language patterns. Despite advancements, detecting hate speech remains challenging due to the evolving nature of language and regional differences. Hate speech can be subtle, context-dependent, and expressed through slang or code words. Additionally, ethical concerns arise with automated moderation, balancing content removal and free speech while avoiding algorithmic bias.

1.3 Problem Statement

The ubiquity of hate speech on internet forums poses serious problems, one of which is the requirement for efficient detection systems in order to preserve polite, inclusive

online communities. It becomes exceedingly difficult to create computers that can recognize hate speech amid language that is always changing and sophisticated. In addition, the ethical and sociological ramifications of automatic content moderation must be addressed.

1.4 Objectives

The goal of the project is to deeply comprehend current strategies, and obstacles to action as well as possibilities to create recognition of hate speech technological innovations. The main goal is to gain an understanding of how to make detection systems more accurate and efficient while taking the moral implications of automated content control into account.

1.5 Scope and Limitations

This research especially examines the techniques and technology used in this area of world-wide debate propagation of hatred identity. Although hate speech is recognized in wider terms on a wide range of digital platforms, forums are the only ones considered because of their particular qualities and difficulties.

A thorough grasp of hate speech detection in social media environments, encompassing cutting-edge methods, difficulties identifying and classifying hate speech, and ethical issues, is the expected result of this project. The project also attempts to pinpoint possible directions for further study and advancement in this area.

1.6 Report Organization

Chapter 1: Introduction to the research study, discussing the problem statement, motivation and objective, and what will be the project outcome.

Chapter 2: Learning about the related research works, discussing briefly, comparison between previous works, then gap analysis.

Chapter 3: Discussion about requirement analysis, data collection, methodology, and overall management of the project and financial analysis.

Chapter 4: Developing the prototype and testing the overall system also discussing the evaluation process.

Chapter 5: Analyzing the result and output we've gotten from this system.

Chapter 6: The overall outcome from this project, and how this project has an impact on real world scenarios, especially for the Bangladesh case.

Chapter 7: Wrapping this project up with a conclusion and what can we do with this project in future. This chapter also has the discussion about limitations.

1.7 Summary

The primary objective of the research aims to tackle the dire need for efficient online discussion forum comments that promote hatred identification. The goal of the research project is to aid in the creation of more reliable and socially conscious automated content moderation systems by reviewing the literature that has already been written, debating methodological strategies, and taking ethical considerations into account.

CHAPTER 2
LITERATURE REVIEW

2.1 Overview

As we are diving into a topic which is one the most sensitive issues of this time, we have read some papers regarding this topic. In this chapter, we will be discussing those papers. It will highlight some important sections like which algorithm they have used, from where they have collected the dataset, how many algorithms are applied on the model and which algorithm has given the best results. And by this, we can have a proper idea of each paper that is vulnerable information for our thesis work.

2.2 Related Works

In the following papers, several algorithms have been applied and by those they have detected hate speech with different accuracy from different algorithms.

Niloofar Safi Samghabadi et.al [1] crawled almost 586k question answer pairs from ask.fm users. Their final data have around 350 question answer pairs. A list which they have made using NUR and given it an interface by using crowdFlower. They have applied Linear SVM for the final accuracy and also some NLP features like POS coloured n grams, Lexical, Emoticons, Sentiwordnet, LIWC, Style and writing density, LDA.They splitted their data into two parts where training data is 70% and testing data is 30%.

David Van Bruwaene et.al [3] have collected around 200 posts for labeling. They have collected the dataset from VISR. In this paper, the applied SVM, CNN,Xbooster. They divided their first experimental dataset into three sections (training 60%,testing 20%,validation 20%) and final result their dataset have the ratio of 80:20 for training and testing model respectively. The examination data for the assault sample involve the SVM, XGBoost, CNN, and the starting point models (ZeroR: Efficiency of 0.835 and a weighted average F-measure of 0.76).

Esraa Omran et.al [4] have collected their data from a famous data science platform named Kaggle and they have applied four algorithms called Naive Bayes,Decision

Tree,SVM,KNN. The accuracy rate was 0.887 and F1_score was 0.885 for both naive bayes and decision tree. The dataset on which they have worked, is the comparison of 24,783 tweets from different users.

Hitesh Kumar Sharma et.al [5] have collected their most of the dataset from twitter, a large source of comments, then they applied the algorithms like Logistic Regression, SVM,Random Forest and Gradient Boost and found the overall testing accuracy about 50-55% and training accuracy 75%-90% for all four classifiers.

Mahep Mahat [6] has taken the data from Twitter,Wikipedia and Formspring. In this paper, SVM,CNN,Naive Bayes, Logistic Regression. Around 3000 sample were applied in the model and the number of testing data was around 9000 examples. The final accuracy of this paper was 77.9%.

2.3 Comparison between existing works

Table 2.3.1. Comparative analysis with previous work

SL No	Author Name	Used Algorithm	Best Accuracy with Algorithm
1.	Niloofar Safi Samghabadi et al. [1]	Linear SVM	AUC=84%
2	Linda Zhou, Andrew Caines, Ildiko Pete, Alice Hutchings[2]	BERT (Bidirectional Encoder Representations from Transformers)	F1_score= 0.69
3.	David Van Bruwaene et.al [3]	SVM, CNN,Xbooster.	SVM=83%
4.	Esraa Omran et.al [4]	Naive bayes,Decision tree	Accuracy= w 0.887 and F1_score= 0.885 for both
5.	Hitesh Kumar Sharma et.al [5]	Logistic Regression, SVM,Random Forest and Gradient Boost	50-55% for all
6.	Mahep Mahat [6]	SVM,CNN,Naive Bayes, Logistic Regression	Naive Bayes=77.9%

2.4 Open Issues

Despite significant progress in hate speech detection, several open issues remain:

1. **Dynamic Nature of Language**: Hate speech evolves rapidly, incorporating new slang, euphemisms, and cultural references, making it difficult for algorithms to keep up.
2. **Data Diversity and Quality**: The variability in datasets, including differences in language, context, and platform-specific behaviors, poses challenges in creating comprehensive models. High-quality, representative datasets are essential but often hard to obtain.
3. **Algorithmic Bias**: Many algorithms can exhibit biases, leading to over- or under-identification of hate speech in certain groups. Ensuring fairness and reducing bias in detection systems is crucial.
4. **Contextual Understanding:** Hate speech is highly context-dependent. Current models often struggle to accurately interpret the context, leading to false positives and negatives.
5. **Ethical and Legal Considerations:** Balancing effective hate speech detection with the protection of free speech and privacy rights is a complex issue that requires careful consideration of ethical and legal frameworks.

2.5 Summary

In this section, we have seen that there are several papers that have tried to talk about this social issue, applied algorithms in order to find the hate words effectively. So, to complete this work with an effective result, we can see that we will have to be more selective about what we are going to do and which challenges we might face while collecting the raw dataset.

CHAPTER 3
METHODOLOGY/ REQUIREMENT ANALYSIS & DESIGN SPECIFICATION

3.1 Overview

In this section, we will talk about the most important part of this report, which is methodology which means how we will try to collect the raw dataset, apply the suitable algorithms on them, in which steps we will extract the dataset.

3.2 Proposed Methodology

As we have a dataset of 7 thousand hate speech and free speech, We will use CNN, RNN, LSTM to see the best outcome among these. As we are partially working on existing data, at first we will apply these steps to make the data usable.

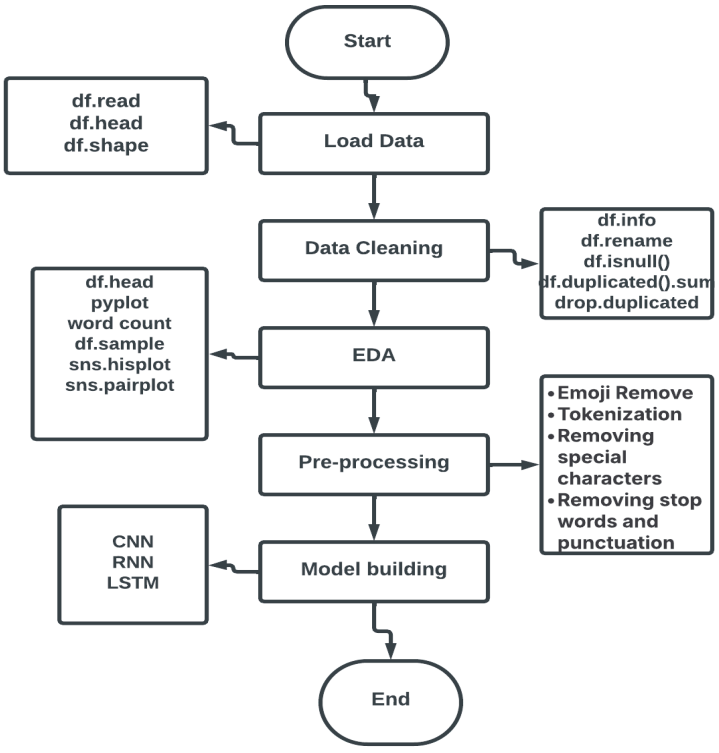


Figure 3.2.1: Process of preparing data for execution

Text input: Obtain the text input from social media sources like facebook,youtube,instagram, plus social networking sites.

Data load: We have loaded the data in the google colaboratory by running codes like df.read(it will read the data from csv file or if we select the option mount drive, then it will fetch the dataset from google drive), df.head(10) by using this command we can shown the first 10 texts of the dataset), df.shape(this command shows the size of the dataset and also the column numbers of the dataset).

index	Comment	Opinion
0	অনুগ্রহ করে বলবেন আপনাদের এই সুপারশপটি কোথায়?	free speech
1	অনুদান পাঠানোর ঠিকানাটা কি দেওয়া যাবে?	free speech
2	অনুপ্রেরণা দিয়ে পাশে থাকার জন্য কৃতজ্ঞতা। দোয়া করবেন যেন শুদ্ধভাবে কাজগুলো করে যেতে পারি	free speech
3	অনেক অনেক দোয়া রইল এত সুন্দর সিদ্দান্ত নিয়ার জন্য এক টাকায় আহার এর সকল ভাইদের কে আল্লাহ পাক মক্কা-মদিনা জিয়ারত নসিব করুক আমিন বিদ্যানন্দের এই কাজটা সফল হোক এই দোয়া করি। এক টাকায় আহার এর পাশে আছি আর সব সময় পাশে থাকবো।	free speech
4	অনেক অনেক দোয়া রইল এত সুন্দর সিদ্দান্ত নিয়ার জন্য এক টাকায় আহার এর সকল ভাইদের কে আল্লাহ পাক মক্কা-মদিনা জিয়ারত নসিব করুক আমিন বিদ্যানন্দের এই কাজটা সফল হোক এই দোয়া করি। এক টাকায় আহার এর পাশে আছি আর সব সময় পাশে থাকবো।	free speech
5	অনেক এক্সপার্ট তাইনা?।	free speech
6	অনেক চেষ্টার পর বয়েস দিয়ে মনের ভাব প্রকাশ করলাম।	free speech
7	অনেক টাকা হলে সাকিব আল হাসানকে ভাড়া করতাম।	free speech
8	অনেক ডাক্তার ইদানীং সুন্দর করে লেখে।	free speech
9	অনেক দিন পরে পুলিশের একটা ভালো কাজের সন্ধান পেলাম। বাফু	free speech

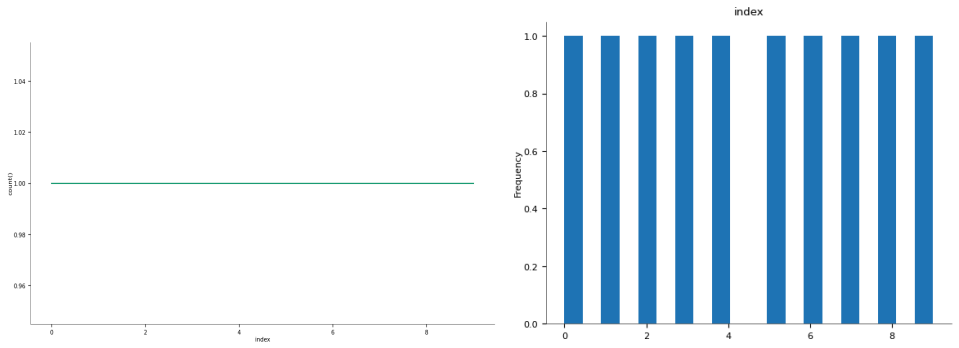


Figure 3.2.2: Distribution & Time Series

Data Cleaning: In the data cleaning process, we have used some functions like df.info(this function identifies the dataframes, rangeindex, data columns,data types and memory usages), df.rename(this function renames the columns names to new names and sample(5) meaning we have shown first 5 sentences of the dataset. After renaming the dataset, we have used Labelencoder to encode the hate speeches into 1 and free speeches into 0. df.is null (), df.duplicated(), functions indicate the null rows and duplicate values of the dataset.

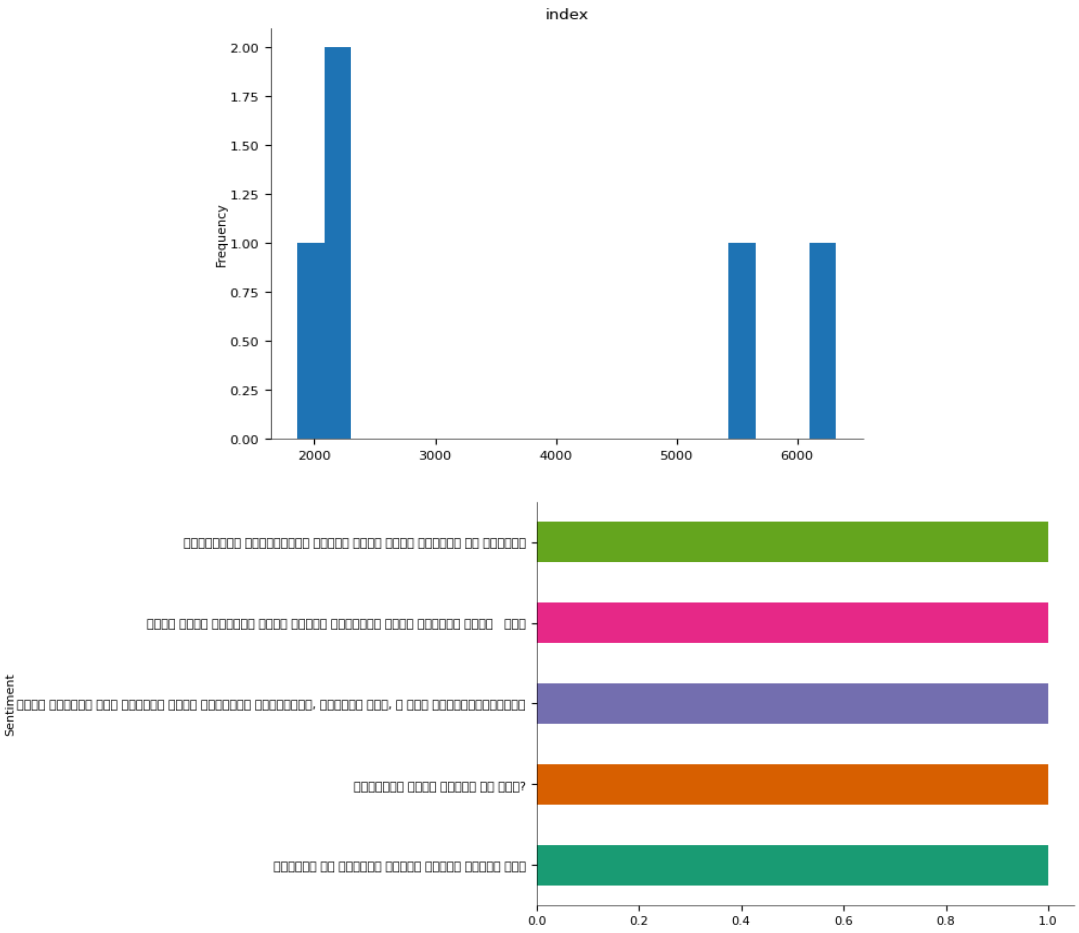


Figure 3.2.3: Categorical Distribution

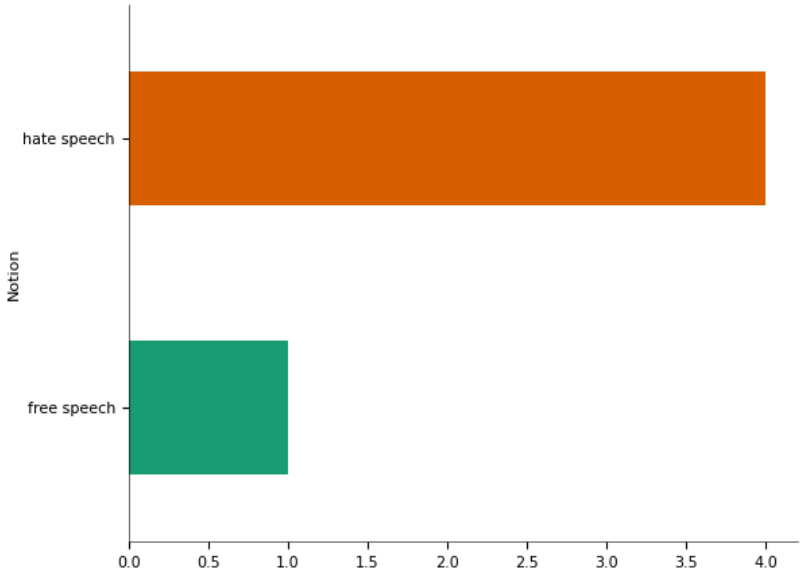


Figure 3.2.4: Categorical Distribution

EDA(Exploratory Data Analysis):

Comprehending Data Distribution: Facilitates comprehension of the data distribution, encompassing measurements of dispersion (variance, standard deviation), and the main trend (mean, median).

Finding Outliers: Looks for anomalies or outliers that could have an impact on the performance of the model.

Finding Missing Values: This process locates empty or vacant values in the dataset that require special attention.

Comprehending Relationships: Examines how various variables relate to one another in order to inform feature engineering and model selection.

Testing Hypotheses: Offers a means of putting preliminary theories regarding the data and its connections to the test.

This part includes a reread of the data, word counts, translations, and stopword searches from our dataset. Pyplot, Hisplot, and Pairplot of our training and testing data were finally obtained.

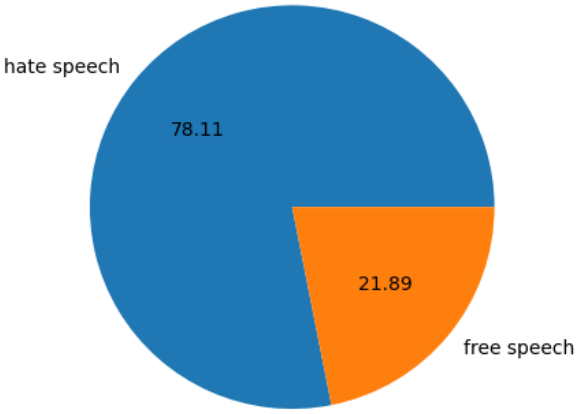


Figure 3.2.5: Pie chart

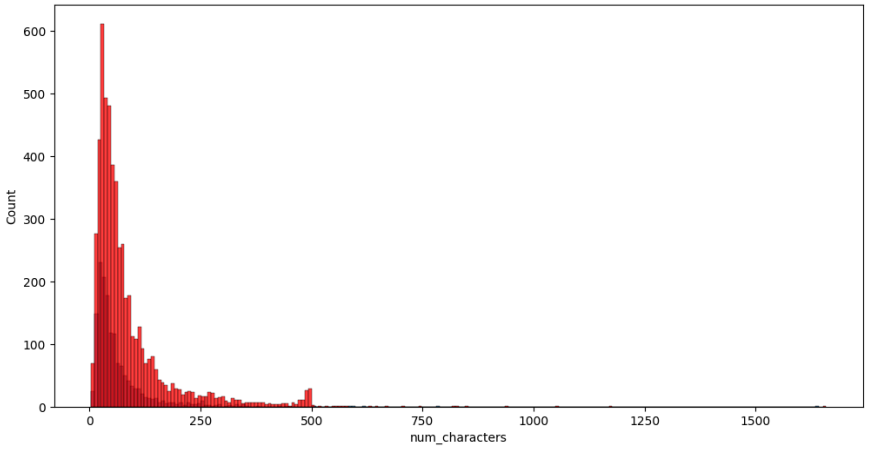


Figure 3.2.6: Histogram of EDA

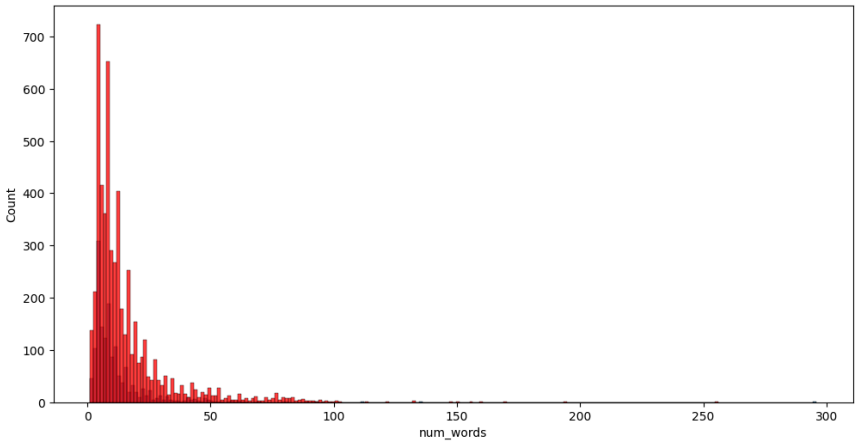


Figure 3.2.7: Histogram of EDA

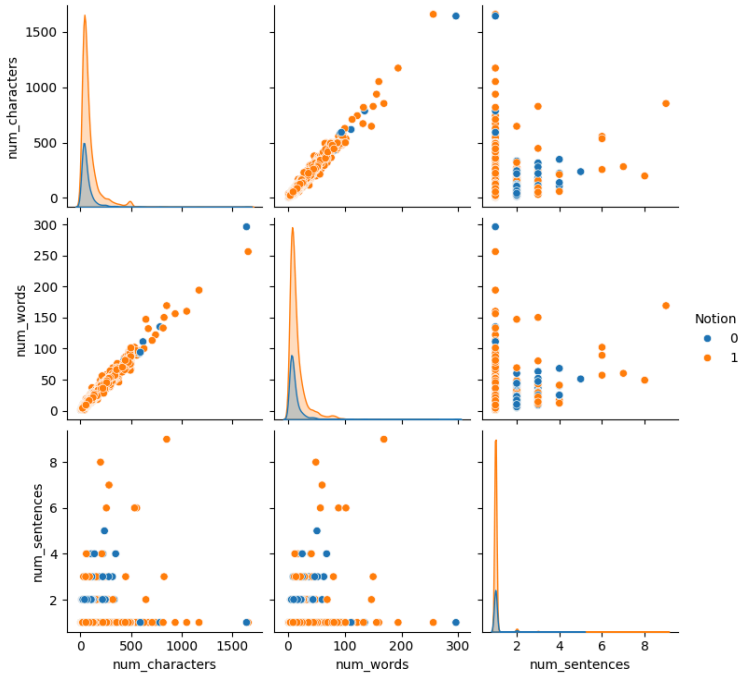


Figure 3.2.8:Pairplot

Preprocessing: In this section, we have removed special characters from our dataset, tokenized them,removed emoticons and removed stopwords and punctuations by using some functions to improve model efficiency.

Model Building: Feed an abuse recognition machine with the retrieved features to make predictions. The model may be built using deep learning architectures (e.g., CNNs, RNNs, LSTM models The possibility that offensive language will be present in the input text is expressed through the label of the class or chance score that the model gives.

Feedback: Explain or provide context for the categorization choice to post-process the discriminatory language that has been found. - Gather input from coordinators or customers to enhance the model and solve new issues or biases.

3.3 Hardware/ Software Requirement

Hardware Requirements:

1. **Processor:** Intel Core i5 or higher

   A powerful processor is essential for efficient computation and faster processing of large datasets.
2. **RAM:** 16 GB or more

   Sufficient memory is crucial for handling extensive data and running complex algorithms without performance bottlenecks.
3. **Storage:** 500 GB SSD

   A solid-state drive (SSD) is recommended for faster data access and storage of large datasets and model outputs.
4. **GPU:** NVIDIA GeForce GTX 1060 or higher

   A dedicated GPU is beneficial for accelerating the training process of Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN).

Software Requirements:

1. **Operating System:** Windows 10, macOS, or Linux (Ubuntu 18.04 or higher)

   The project can be developed on any of these operating systems, ensuring compatibility with necessary software tools.
2. **Programming Language:** Python 3.7 or higher

Python is the primary language for implementing machine learning and NLP models due to its extensive libraries and community support.

3. **IDE/ Code Editor:** PyCharm, Google Colaboratory, VSCode
   An integrated development environment (IDE) or code editor is required for writing, testing, and debugging code.
4. **Libraries and Frameworks:**
   - TensorFlow/Keras: For building and training RNN and CNN models.
   - NLTK/Spacy: For natural language processing tasks.
   - Pandas/Numpy: For data manipulation and analysis.
   - Matplotlib/Seaborn: For data visualization.
5. **Dataset:** Stored in Google Spreadsheets.
   A database is required for storing and managing the collected datasets.

6. **Version Control System:** Git
   Git is essential for tracking changes in the codebase and collaborating with team members.
7. **Virtual Environment:** Anaconda or venv
   Using a virtual environment helps manage dependencies and ensures consistency across different development environments.

3.4 Project Management and Financial Analysis

At first, we made our own dataset from different social media platform. Then uploaded it to google colaboratory, read the csv file, imported library function like matplotlib,pandas,seaborn,numpy then we will the divide dataset into training data and testing data where the number of training model will be large than testing model.

- ## Project Objective

  1. To Develop a machine learning or deep learning model that accurately identifies hate speech in text, photos and other multimedia content across internet platforms.
  2. To refine and optimize hate speech detection algorithms to reduce false positives and negatives.

3. To provide a platform for farmers to sell their product directly to their customers
4. To adapt hate speech detection to many languages and cultures, taking into account linguistic intricacies, slang and geographical differences.

- **Project Timeline**

1. Phase 1: Project Planning and Background Research (2-3 weeks)
2. Phase 2: Design and Prototype Implementation (5-7 weeks)
3. Phase 3: Dataset collection and preprocessing (8-12 weeks)
4. Phase 4: Testing (2 weeks)
5. Phase 5: Deployment (ongoing)

- **Resource Planning**

1. **Equipment and Tools**
    - **a.** Development and Testing Servers
    - **b.** High-performance Computers for Development Team Design Software (Figma)
    - **c.** Collaboration Tools (Google colaboratory, Visual Studio,cloudFlower)
2. **Data and Content**
    - **a.** Data: We have collect dataset manually from social media platforms
    - **b.** Images: We will try to collect the images from social media.
    - **c.** User Documentation: After collecting the dataset, we will write the documentation

3. **Training and Skill Development**
    a. For completing the thesis based project, we have to present a demo website, so we will take a course online based on html,css so that we can work effectively.

- ## Communication Plan

    1. **Stakeholder Meetings:**

        Purpose: Updating project requirements gathering process and gathering feedback from stakeholders.
        Participants: Team members, stakeholders.
        Frequency: Bi-weekly.

    2. **Change Control Meetings:**

        Purpose: Discussing any changes in the project.
        Participants: Project supervisor and team members.
        Frequency: Weekly or as needed when change requires.

Financial Analysis:

While downloading the dataset from Kaggle, some megabytes have been spent (wifi cost), which costs around 200 tk.

Table 3.4.1. Estimated Cost for this project

SN	Components	Estimated Cost (BDT)
01	Software and Tools	1,500
02	Domain and Hosting	10,000
03	Data Collection and Processing	2,500
04	Training for Learning New Technology	10,000
05	Documentation and Report Writing	1,000
	Total Estimated Cost:	25,000

3.5 Summary

Hate speech detection using Natural Language Processing (NLP) techniques has developed as an important topic of research, to reduce the negative consequences of hate speech on online platforms and promote a safer digital environment. Researchers have created a nseveralber of ways to autonomously identify and classify hate speech in written material, using advances in computer learning and linguistic analysis. Hate speech identification using NLP approaches is a promising approach to combating online hate speech while also building a more inclusive and polite online community. Investigators use linguistic analysis and machine learning to increase the state-of-the-art in hate speech identification, tackling critical problems and opening the way for more effective mitigation techniques.

CHAPTER 4
IMPLEMENTATION

4.1 Overview

In this chapter, we will discuss how we trained the model, prepared the model before the final testing, and how the models were evaluated when they gave the final result. Collecting and preparing a labeled dataset—text samples identified as offensive language or not—is necessary for the NLP-based hate speech detection process. Word embeddings are employed to turn the text into numeric representations after cleaning, tokenization, lowercasing,stop word removal, stemming/lemmatization, and other text preprocessing techniques. Both CNN and RNN architectures are used for the model. While the RNN model employs LSTM or GRU layers to capture sequential dependencies in the text, the CNN model utilizes convolutional layers and maximum pooling to identify local text patterns.Both models have layers of output that are completely interconnected. They have been optimized with strategies similar to Eve and then trained using a binary loss of cross-entropy rate. Accuracy is used to assess the performance of the model.Ultimately, a comparison is conducted between the algorithms to ascertain whether one performs better overall and with more precision in identifying hate speech.

4.2 Train Model

Before training the model, we annotated the dataset by 7 people. We developed models utilizing Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) and LSTM(Long Short-Term Memory) in order to achieve accurate classification in the use of NLP for hateful language identification. We created and trained our CNN, RNN and LSTM models by gathering a labeled dataset and preparing the text by tokenizing, cleaning, and turning it into numbers using word embeddings. Convolutional and max-pooling layers were employed by the CNN algorithm to extract local features from the text, and fully connected layers were then used to carry out the final classification. In contrast, the sequential nature of the text data was captured by the RNN model through the use of LSTM layers. The two-dimensional cross-entropy loss function was used to train both models, and the Adam optimizer was used to optimize the results. We tracked performance parameters

like accuracy in the training phase. Ultimately, we identified the best-performing method for this important task through contrasting the assessment findings and determining the accuracy and overall efficacy of every model in detecting statements of hatred. We have trained our model by using CNN,RNN,and LSTM. And the hate comments have several emoticons, punctuation marks and also some symbols that can not be identified by the machines , so we have proposed a model where all the emoticons, symbols will be replaced or deleted with the text.Also, we have preprocessed the dataset that we have collected from the several social platforms. We are working on the code and also trying to add more algorithms to train the model.

The architecture of recurrent neural networks (RNNs):

Convolutional Layers: These layers send the result of a convolution operation to the subsequent layer after performing it to the input. Convolutions use tiny rectangles of input data to learn visual attributes and maintain the distances between pixels.

Kernels/filters: Tiny matrices that are used to identify characteristics such as edges, textures, etc.

Stride: The filter's movement measured in pixels.

To manage the spatial size of the output, padding is the addition of pixels around the input image.

ReLU (Rectified Linear Unit) functions are typically used as activation functions to introduce non-linearity.

Pooling Layers: These layers preserve the most important information while reducing the dimensionality of each feature map. The most popular kind, known as max pooling, uses the highest value found in each feature map sub-region.

Fully Connected Layers: Fully connected layers perform high-level reasoning after a number of convolutional and pooling layers. Every activation in the layer above is coupled to every neuron in a completely connected layer.

The output layer produces the final result and, in the case of classification problems, typically includes a softmax activation function.

Uses:

Recognition of images and videos

Image categorization

Object recognition,analysis of medical image

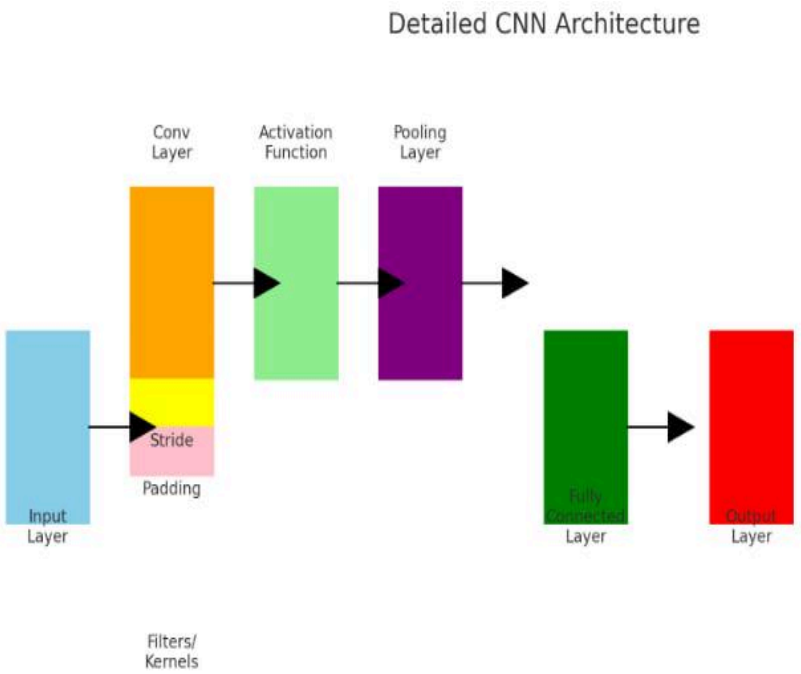


Figure 4.2.1: CNN Architecture

The architecture of recurrent neural networks (RNNs):

Recurrent Layers: Recurrent neural networks (RNNs) can preserve a state across input sequences because, in contrast to feedforward neural networks, they feature connections that form directed cycles.

RNNs possess a secret state that is updated every time step.

Input Sequence: RNNs process one element of an input sequence at a time after receiving it as a sequence.

Use activation functions such as ReLU or Tanh as a general rule.

Output Layer: Depending on the task (e.g., sequence-to-sequence tasks vs. sequence classification), the output may be a sequence or a single value.

Uses:

Forecasting time series

Text production and language modeling

automated translation,Recognition of Speech

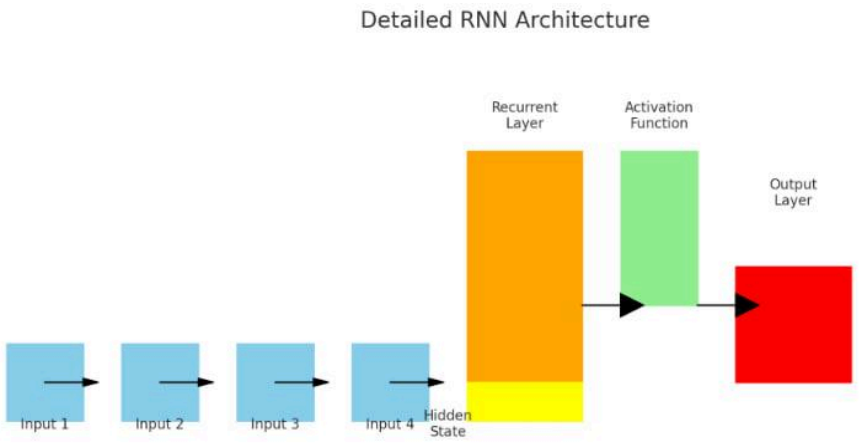


Figure 4.2.2: RNN Architecture

Architecture for Long Short-Term Memory (LSTM):

LSTM Cells: Using unique units known as LSTM cells, LSTM networks are a kind of RNN created to get around the long-term dependency issue.

Cell State: This is the network's memory, which is updated continuously.

Gates:

The Forget Gate selects which data from the cell state should be deleted. The input gate determines what additional data should be added to the cell state. The output gate selects the data to output in accordance with the state of the cell. Use activation functions such as sigmoid and tanh within the gates, as this is the standard procedure.

Layers: LSTM layers may be stacked on top of one another.

Uses:

Sequence forecasting

Text production and language modeling

Forecasting time series

Recognition of Speech and Handwriting

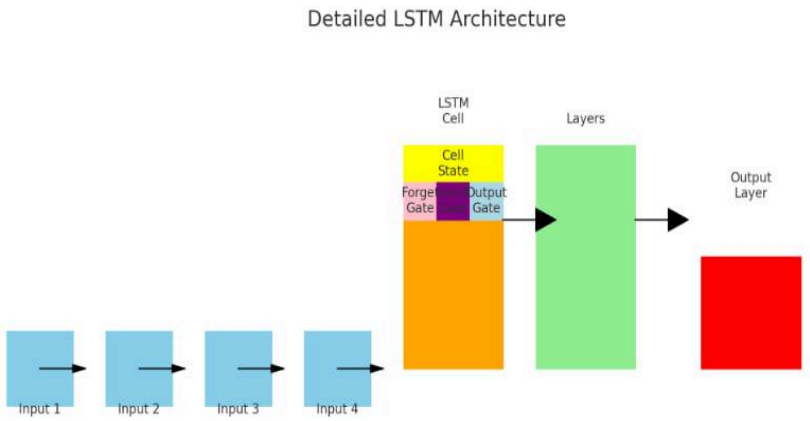


Figure 4.2.3: LSTM Architecture

These architectures are selected according to the particular needs of the work at hand, and they serve as the foundation for a large number of contemporary machine learning applications.

4.3 Model Evaluation

In this section, we have taken 5934, and the testing set is 1484. After that, we took a sentence from the training data, and we have also checked the correctness of the data by debugging it which leads to the authentication and verification of the given data samples and corresponding data labels. To evaluate it with more authenticity, we used tokenizer as well to check the word index length. We validated the data where the epochs were 5. For real-time applications, the effectiveness of the detection process—including its speed and resource usage—is taken into account to maximize performance. Lastly, the system's capacity to evolve and grow over time—often via constant learning and updates—improves its efficacy in stifling hate speech in a variety of online contexts.

4.4 Summary

Using natural language processing (NLP) for hate speech identification in online forums entails a thorough procedure that includes data preparation, model training, and system testing. The first step in the overview is to compile a labeled dataset of forum postings and identify any that include hate speech. After preprocessing (cleaning, tokenization, lowercasing, stop word removal, stemming/lemmatization), embeddings of words are used to transform the text into numerical representations. Both recurrent neural networks, or RNN, and convolutional neural network networks (CNN) are used for model training. While the model based on R uses layers of LSTM to capture consecutive dependencies, the CNN model uses layers of convolution and pooling to capture local patterns. Accuracy is used to track performances for both models, which have been optimized using the Adam optimizer after being trained with a dual-entropy function of loss.

CHAPTER 5
RESULT AND ANALYSIS

5.1 Overview

In this section, we will discuss the results that we have got from the testing set,then we will analyze the results according to the models and also compare those with the respective models and also will be sharing the best accuracy algorithms that have given the highest accuracy.

5.2 Experimental Result

As we have used two models CNN (Convolutional Neural Network),RNN (Recurrent Neural Network) to get the accuracy. So,now we will share the CNN, RNN accuracy levels.

CNN:

```
Epoch 1/10
186/186 [==============================] - 17s 78ms/step - loss: 0.4253 - accuracy: 0.8045 - val_loss: 0.3237 - val_accuracy: 0.8524
Epoch 2/10
186/186 [==============================] - 14s 78ms/step - loss: 0.1453 - accuracy: 0.9491 - val_loss: 0.3387 - val_accuracy: 0.8794
Epoch 3/10
186/186 [==============================] - 14s 77ms/step - loss: 0.0270 - accuracy: 0.9924 - val_loss: 0.4386 - val_accuracy: 0.8666
Epoch 4/10
186/186 [==============================] - 14s 77ms/step - loss: 0.0090 - accuracy: 0.9985 - val_loss: 0.4943 - val_accuracy: 0.8673
Epoch 5/10
186/186 [==============================] - 14s 77ms/step - loss: 0.0064 - accuracy: 0.9987 - val_loss: 0.5130 - val_accuracy: 0.8693
Epoch 6/10
186/186 [==============================] - 14s 77ms/step - loss: 0.0055 - accuracy: 0.9987 - val_loss: 0.5516 - val_accuracy: 0.8699
Epoch 7/10
186/186 [==============================] - 14s 77ms/step - loss: 0.0057 - accuracy: 0.9987 - val_loss: 0.5035 - val_accuracy: 0.8740
Epoch 8/10
186/186 [==============================] - 14s 76ms/step - loss: 0.0018 - accuracy: 0.9992 - val_loss: 0.5563 - val_accuracy: 0.8699
Epoch 9/10
186/186 [==============================] - 14s 77ms/step - loss: 0.0013 - accuracy: 0.9993 - val_loss: 0.6130 - val_accuracy: 0.8760
Epoch 10/10
186/186 [==============================] - 14s 76ms/step - loss: 0.0021 - accuracy: 0.9990 - val_loss: 0.6113 - val_accuracy: 0.8780
47/47 [==============================] - 1s 11ms/step - loss: 0.6113 - accuracy: 0.8780
CNN Test Accuracy: 87.80%
```

RNN:

```
Epoch 1/10
186/186 [==============================] - 17s 84ms/step - loss: 0.5374 - accuracy: 0.7873 - val_loss: 0.5500 - val_accuracy: 0.7729
Epoch 2/10
186/186 [==============================] - 15s 81ms/step - loss: 0.5243 - accuracy: 0.7914 - val_loss: 0.5433 - val_accuracy: 0.7729
Epoch 3/10
186/186 [==============================] - 15s 82ms/step - loss: 0.5324 - accuracy: 0.7897 - val_loss: 0.5418 - val_accuracy: 0.7729
Epoch 4/10
186/186 [==============================] - 15s 81ms/step - loss: 0.5497 - accuracy: 0.7799 - val_loss: 0.5408 - val_accuracy: 0.7729
Epoch 5/10
186/186 [==============================] - 15s 82ms/step - loss: 0.5334 - accuracy: 0.7905 - val_loss: 0.5436 - val_accuracy: 0.7729
Epoch 6/10
186/186 [==============================] - 16s 84ms/step - loss: 0.5262 - accuracy: 0.7927 - val_loss: 0.5370 - val_accuracy: 0.7729
Epoch 7/10
186/186 [==============================] - 17s 90ms/step - loss: 0.5191 - accuracy: 0.7931 - val_loss: 0.5381 - val_accuracy: 0.7729
Epoch 8/10
186/186 [==============================] - 16s 83ms/step - loss: 0.5185 - accuracy: 0.7932 - val_loss: 0.5420 - val_accuracy: 0.7729
Epoch 9/10
186/186 [==============================] - 15s 83ms/step - loss: 0.5184 - accuracy: 0.7932 - val_loss: 0.5353 - val_accuracy: 0.7729
Epoch 10/10
186/186 [==============================] - 16s 84ms/step - loss: 0.5179 - accuracy: 0.7932 - val_loss: 0.5418 - val_accuracy: 0.7729
47/47 [==============================] - 1s 12ms/step - loss: 0.5418 - accuracy: 0.7729
RNN Test Accuracy: 77.29%
```

LSTM:

```
Epoch 1/10
186/186 [==============================] - 63s 285ms/step - loss: 0.4068 - accuracy: 0.8224 - val_loss: 0.3437 - val_accuracy: 0.8464
Epoch 2/10
186/186 [==============================] - 48s 258ms/step - loss: 0.1942 - accuracy: 0.9312 - val_loss: 0.3298 - val_accuracy: 0.8713
Epoch 3/10
186/186 [==============================] - 48s 259ms/step - loss: 0.0515 - accuracy: 0.9855 - val_loss: 0.3467 - val_accuracy: 0.8821
Epoch 4/10
186/186 [==============================] - 48s 258ms/step - loss: 0.0189 - accuracy: 0.9953 - val_loss: 0.4355 - val_accuracy: 0.8881
Epoch 5/10
186/186 [==============================] - 48s 258ms/step - loss: 0.0085 - accuracy: 0.9983 - val_loss: 0.4930 - val_accuracy: 0.8807
Epoch 6/10
186/186 [==============================] - 50s 267ms/step - loss: 0.0079 - accuracy: 0.9987 - val_loss: 0.5558 - val_accuracy: 0.8848
Epoch 7/10
186/186 [==============================] - 49s 263ms/step - loss: 0.0072 - accuracy: 0.9988 - val_loss: 0.5500 - val_accuracy: 0.8821
Epoch 8/10
186/186 [==============================] - 49s 262ms/step - loss: 0.0072 - accuracy: 0.9988 - val_loss: 0.6339 - val_accuracy: 0.8760
Epoch 9/10
186/186 [==============================] - 48s 260ms/step - loss: 0.0070 - accuracy: 0.9988 - val_loss: 0.5859 - val_accuracy: 0.8807
Epoch 10/10
186/186 [==============================] - 48s 261ms/step - loss: 0.0074 - accuracy: 0.9988 - val_loss: 0.5174 - val_accuracy: 0.8854
47/47 [==============================] - 3s 70ms/step - loss: 0.5174 - accuracy: 0.8854
Test Accuracy: 88.54%
```

Training and testing accuracy:

The dataset contains 7,418 rows. To determine the number of training and testing samples, we need to apply a train-test split. Typically, a common split is 80% for training and 20% for testing.

Let's calculate the number of training and testing samples:

Training Data (80%): $0.8 \times 7418 \approx 5934$

Testing Data (20%): $0.2 \times 7418 \approx 1484$

So, the dataset will be split into approximately 5,934 training samples and 1,484 testing samples.

CNN & RNN Accuracy:

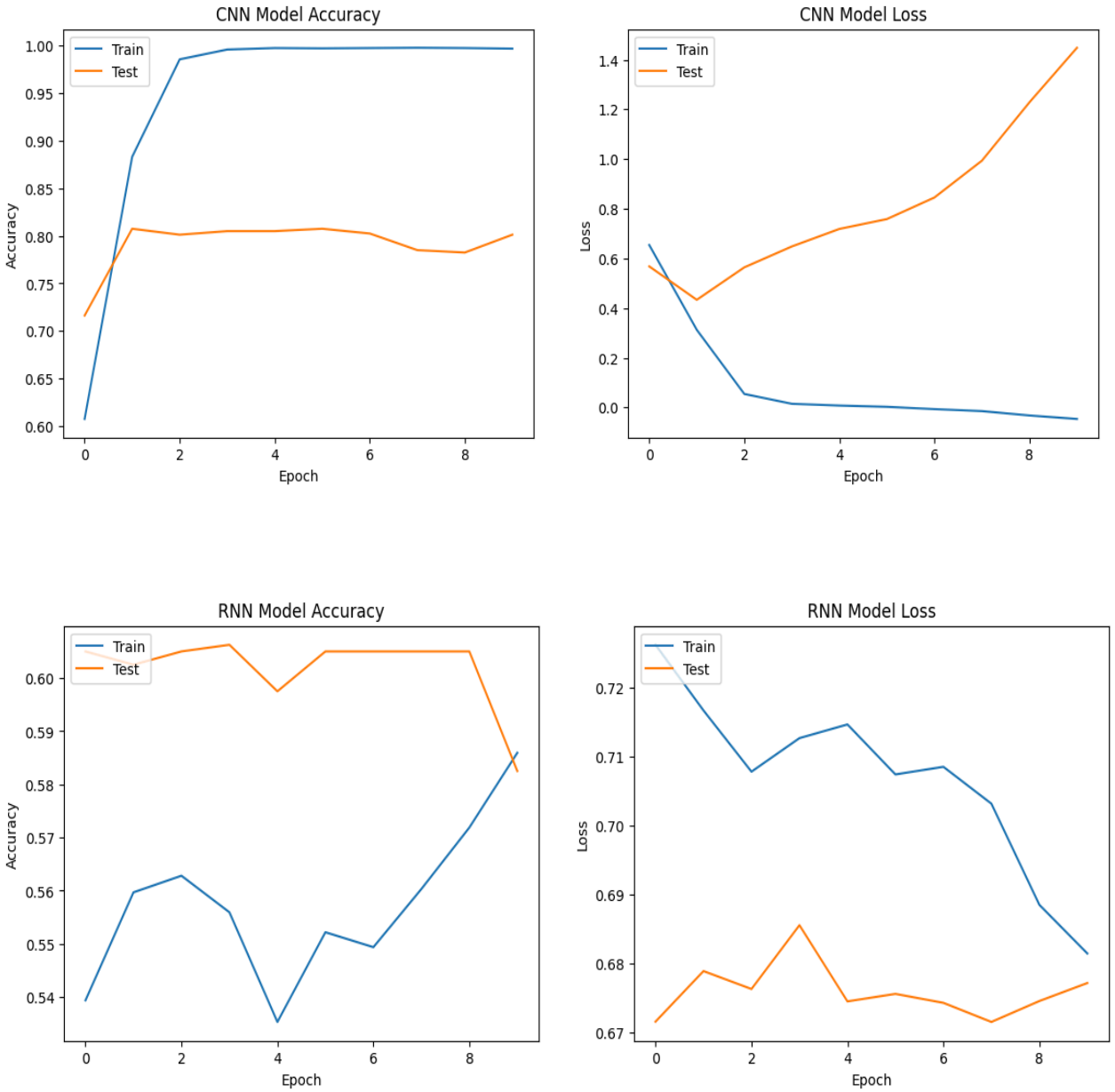


Figure 5.2.1: RNN, CNN Comparison

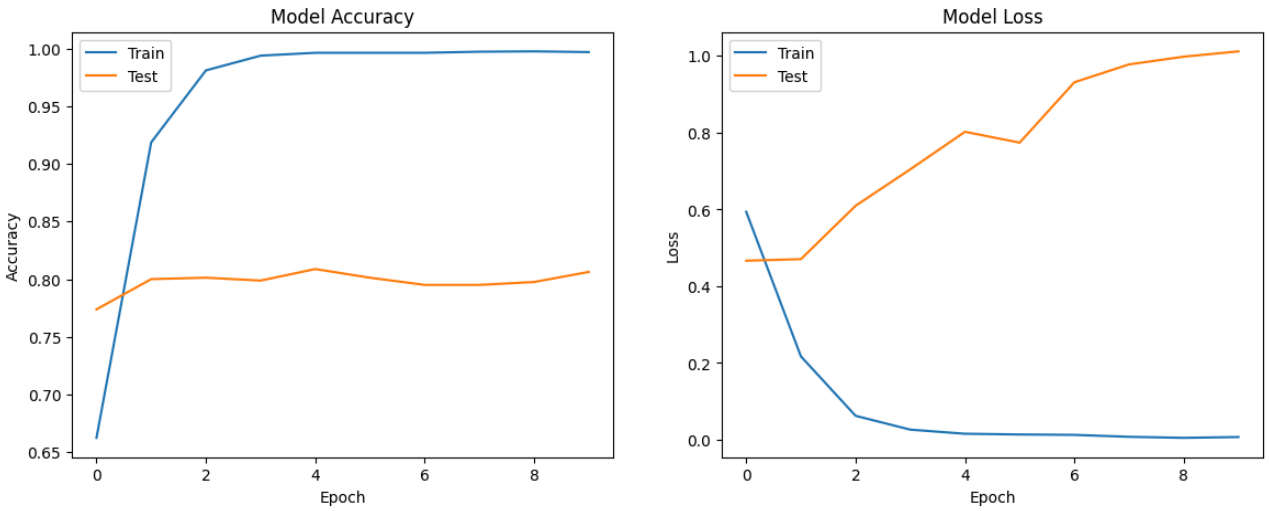


Figure 5.2.2: Model Loss

From the graphs, we can see that we have got 87.80% accuracy from CNN, 77.29% accuracy from RNN model and LSTM 88.54%. We can see that LSTM has better accuracy than the other two models.

5.3 Performance Analysis

This comparison is all about to find out the best performance among the algorithms that we have applied in the thesis.

- **CNN (Convolutional Neural Network):** By applying this algorithm, we have got 87.80% of accuracy on our dataset.
- **RNN(Recurrent Neural Network) :** By applying this algorithms, we have got 77.29% accuracy on our dataset.
- **LSTM(Long Short-Term Memory) :** By applying this algorithms, we have got 88.54% accuracy on our dataset.

5.4 Summary

From the whole result analysis, we can clearly see that, LSTM model works more effectively than the CNN & RNN model.

CHAPTER 6
IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY

6.1 Impact on Life

By tackling the widespread problem of online hate speech, the development of hate speech detection utilizing techniques based on Natural Language Processing (NLP) has had an important effect on society and digital life. Online forums play a vital role in today's linked world as critical spaces for civic engagement and public conversation. But the unbridled spread of hate speech erodes these areas, creating poisoning, polarization, and even physical violence. Through the use of natural language processing (natural language processing), scientists and programmers are developing tools that can automatically detect and filter hate speech, making the internet a safer and more welcoming place. These technologies help platform administrators to protect user welfare and community requirements while also improving the efficiency of content control. Furthermore, the application of sophisticated machine learning models and linguistic analysis to the identification of hate speech enhances algorithmic accuracy while simultaneously advancing current discussions about moral implications of automated content regulation. As these technologies advance, they have the capacity to suppress hate speech while simultaneously promoting positive discourse, polite interactions, and, in the end, the development of more compassionate and supportive online communities worldwide. Thus, the continued development and use of NLP in hate speech detection represents a critical step in using technology for good social impact and furthering the goals of digital safety and diversity.

6.2 Impact on Society & Environment

The creation and application of sophisticated technology such as Natural Language Processing (NLP) to detect hate speech has significant consequences for the digital world and society. These systems are essential for protecting online communities because they detect and remove harmful content and language that can be discriminatory. By automatically reporting hate speech, these technologies help to promote safer internet experiences for people all around the world by encouraging a

more civil and inclusive digital discussion. In addition, the introduction of these technologies signifies a proactive approach towards tackling social concerns, promoting conscientious digital citizenship, and establishing guidelines for moral conduct on the internet. The ability of hate speech detection systems to lessen the psychological harm and emotional discomfort brought on by exposure to hateful content has an impact on the environment that extends beyond societal gains. These solutions lessen the negative environmental consequences of toxic online interactions by limiting the visibility and transmission of hate speech, promoting a better online ecosystem for all users. Therefore, in the digital age, integrating cutting-edge technical solutions that not only improves digital safety but also favorably impacts societal well-being and the sustainability of the environment.

6.3 Ethical Aspect

The future of online conversation and public safety will be greatly influenced by ethical issues with automated content filtering and hate speech identification. As hate speech detection and suppression technologies improve, a number of moral dilemmas arise that should be carefully considered. First, the debate over speech rights vs regulation of content raises serious concerns. While it is the duty of platforms to provide inclusive and safe spaces, too stringent content filtering practices run the risk of suppressing free speech and a range of opinions. Understanding the subtle differences in legal, social, and cultural norms between various communities and places is necessary to decide when to draw the boundary between damaging speech and acceptable conversation.Second, there is a significant ethical conundrum raised by the possibility of algorithmic bias. The datasets used to train machine learning models for hate speech detection may contain societal biases, thereby sustaining injustices and increasing systemic prejudices. For example, speech from marginalised groups or dialects that depart from the mainstream may be flagged by automated systems disproportionately, which could result in biassed effects. In order to address prejudice in these algorithms, openness in model building is required, along with thorough testing to ensure fairness and continual modifications to reduce unforeseen consequences.

Furthermore, there are ethical questions about the accountability of content filtering algorithms. Frequently without direct human monitoring, automated systems decide

what hate speech is and how to treat it. This lack of transparency can make it difficult to understand how decisions are made and restrict the options available to people who have been unfairly or mistakenly moderated. Maintaining the values of justice and accountability requires maintaining moderation procedures transparent and offering channels for appeal and rectification.

Dependence on automated systems for content control also has wider societal ramifications. Although the goal of these technologies is to create safer online spaces, it's possible that they will unintentionally abdicate the onus of addressing hate speech through proactive communication, community involvement, and education on the part of platform operators and users. Fostering a sustainable and inclusive online culture requires striking a balance between technology solutions and all-encompassing community-driven initiatives.

In conclusion, traversing challenging ethical terrain is necessary for the establishment of efficient hate speech detection tools. It is essential to uphold the values of inclusivity, justice, accountability, and freedom of expression when using technology to effectively counteract hate speech. We may work towards developing online environments that are secure and encouraging of various viewpoints and voices by including stakeholders from a variety of industries and giving ethical concerns top priority through the design and development of these systems.

6.4 Sustainability Plan

The project's ongoing evolution and adaptability to social needs and advances in technology highlight its long-term viability. The ethical implications of bias mitigation and openness in the use of algorithms guarantee the appropriate implementation of hate speech detection systems. Scalable and effective natural language processing (NLP) algorithms that can handle an array of linguistic patterns or intensive processing requirements for data promote technology sustainability. In order to promote stability including scalability, financial sustainability is addressed with inexpensive options for data gathering, handling, and system maintenance. Partnerships with stakeholders, support for online safety, and education campaigns on the effects of hate speech and the role of technology in creating inclusive digital

communities are some of the ways that social sustainability is promoted. In conclusion, the research on natural language processing (NLP) for hate speech identification in online forums seeks to both combat hate speech and maintain environment sustainable development principles through ethical technology implementation with effective application.

6.5 Summary

To sum up, this research has explored the crucial field of identifying hate speech in online forums using Natural Language Processing (NLP) methods in an effort to tackle the urgent need to create more inclusive and secure digital spaces. The widespread use of discriminatory language on online platforms presents serious risks to personal safety and societal cohesiveness, calling for the development of cutting-edge technology solutions. By means of an extensive examination of extant literature and methodology, this study has brought to light the various techniques and algorithms employed in the detection of hate speech.

CHAPTER 7
CONCLUSION AND FUTURE WORK

7.1 Conclusions

To sum up, this research has explored the crucial field of identifying hate speech in online forums using Natural Language Processing (NLP) methods in an effort to tackle the urgent need to create more inclusive and secure digital spaces. The widespread use of discriminatory language on online platforms presents serious risks to personal safety and societal cohesiveness, calling for the development of cutting-edge technology solutions. By means of an extensive examination of extant literature and methodology, this study has brought to light the various techniques and algorithms employed in the detection of hate speech. Every method, from advanced neural network models like CNNs,RNNs or BERT to simple linear SVM, has its own advantages and skills for recognising and categorizing nasty content. The comparative examination of multiple research highlighted the fluctuations in algorithmic performance and the continuous pursuit of increased precision and dependability. Looking ahead, a number of directions for more research and development are suggested by the study's conclusions and insights. To keep up with the constantly changing landscape of online hate speech, it is imperative to continuously improve real-time detection capabilities, integrate multilingual and culturally sensitive techniques, and develop algorithms.

7.2 Further Suggested Works

In future, there can be a huge changes of research by working on some specific classifiers, the researchers can add more options like, to whom the hate speech are being delivered, the weight of the hate comment, the hate speech are being delivered to one person or a community, these annotation can be applied as well to get the best result out of any dataset.

1. **Dataset Expansion**: Continuously updating and expanding the dataset to include new types of hate speech.

2. **Collaborative Research**: Encouraging collaborative efforts among researchers, industry practitioners, and academic institutions to foster innovation and share knowledge.

7.3 Limitations/ Conflict of Interests

Representativeness and Data Bias: The possible bias in the training data that NLP models employ is one major drawback. Strongly dependent on annotated datasets, hate speech detection methods might not fully capture all manifestations of hate speech in a range of language, cultural, and demographic circumstances. The efficiency of the simulation may be impacted by this bias, which may result in errors, underrepresentation of specific marginalized groups, or linguistic variances.

Equity and Algorithmic Bias: Based on the traits and qualities they pick up from the data, NLP algorithms themselves may create biases. Models might unintentionally favor some hate speech over others, for example, or incorrectly identify subtle utterances that deviate from the training set. To guarantee fair and equitable results in hate speech detection, addressing these biases calls for constant inspection and modification.

Difficulties with Contextual Understanding: Hate speech frequently contains nuanced details and contextual dependencies that make it difficult for NLP models to read correctly. Irony, sarcasm, cultural allusions, and regional accents can change a word or phrase's meaning, which can cause false positives or misunderstandings when it comes to hate speech identification.

Language Is Dynamic: Language changes quickly, particularly in online forums where new slang, memes, and idioms are constantly created. In order to preserve relevance and accuracy over time, NLP models trained on static datasets may find it difficult to keep up with these adjustments, requiring ongoing updates and modifications.

Consent from Users and Privacy: Privacy concerns arise when gathering and evaluating user-generated content for hate speech detection; ensuring adherence to

data protection laws and gaining informed consent from users whose data is used for training or assessment is essential but difficult, particularly in large-scale datasets.

Legal and Ethical Issues: Automated content moderation systems, such as those that identify hate speech, present difficult moral conundrums. If algorithms' decisions are not properly implemented and controlled, users may experience negative outcomes in daily life, such as the removal of content or the suspension of their accounts, which could violate their right to free speech.

Conflict of Interest in Platform Moderation: Online forum hosting platforms may have a stake in the identification and management of hate speech. Conflicts over whether to prioritize content filtering over fostering free speech and variety of opinion can arise from the need to strike a balance between user safety, open discourse, and community standards.

It takes a multidisciplinary approach integrating knowledge from the fields of languages, ethics, law, and computer science to navigate these constraints and conflicts of interest. To reduce dangers and promote the ethical use of NLP technology in hate speech detection on online forums, transparency, accountability, and ongoing stakeholder interaction are crucial.

7.4 Summary

This research delves into the critical task of detecting hate speech in online forums using NLP techniques to foster safer digital environments. The study highlights various methodologies and algorithms, noting the strengths and challenges of each. It underscores the necessity for continual improvement in real-time detection, the integration of multilingual approaches, and cultural sensitivity to keep pace with evolving online language. Addressing these challenges requires a multidisciplinary approach and ongoing collaboration among stakeholders to ensure fair and effective hate speech detection.

REFERENCE

[1] N. Safi Samghabadi, S. Maharjan, A. Sprague, R. Diaz-Sprague, and T. Solorio, “Detecting nastiness in social media,” Proceedings of the First Workshop on Abusive Language Online, 2017. doi:10.18653/v1/w17-3010

[2] D. Van Bruwaene, Q. Huang, and D. Inkpen, “A multi-platform dataset for detecting cyberbullying in social media,” Language Resources and Evaluation, vol. 54, no. 4, pp. 851–874, Apr. 2020. doi:10.1007/s10579-020-09488-3

[3] J. Salminen et al., “Developing an online hate classifier for multiple social media platforms,” Human-centric Computing and Information Sciences, vol. 10, no. 1, Jan. 2020. doi:10.1186/s13673-019-0205-6

[4] M. Mahat, “Detecting cyberbullying across multiple social media platforms using Deep Learning,” 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Mar. 2021. doi:10.1109/icacite51222.2021.9404736

[5] H. Kumar Sharma, K. Kshitiz, and Shailendra, “NLP and machine learning techniques for detecting insulting comments on social networking platforms,” 2018 International Conference on Advances in Computing and Communication Engineering (ICACCE), Jun. 2018. doi:10.1109/icacce.2018.8441728

[6] Y. Pandey, M. Sharma, M. K. Siddiqui, and S. S. Yadav, “Hate speech detection model using bag of words and naïve Bayes,” Advances in Data and Information Sciences, pp. 457–470, 2022. doi:10.1007/978-981-16-5689-7_40

[7] D. Mody, Y. Huang, and T. E. Alves de Oliveira, “A curated dataset for hate speech detection on social media text,” Data in Brief, vol. 46, p. 108832, Feb. 2023. doi:10.1016/j.dib.2022.108832

[8] P. Kapil and A. Ekbal, “A deep neural network based multi-task learning approach to hate speech detection,” Knowledge-Based Systems, vol. 210, p. 106458, Dec. 2020. doi:10.1016/j.knosys.2020.106458

[9] R. Ali, U. Farooq, U. Arshad, W. Shahzad, and M. O. Beg, “Hate speech detection on Twitter using transfer learning,” Computer Speech & Language, vol. 74, p. 101365, Jul. 2022. doi:10.1016/j.csl.2022.101365

[10] E. W. Pamungkas, V. Basile, and V. Patti, “A joint learning approach with knowledge injection for zero-shot cross-lingual hate speech detection,” Information Processing & Management, vol. 58, no. 4, p. 102544, Jul. 2021. doi:10.1016/j.ipm.2021.102544

[11] M. Subramanian, V. Easwaramoorthy Sathiskumar, G. Deepalakshmi, J. Cho, and G. Manikandan, “A survey on hate speech detection and sentiment analysis using machine learning and Deep Learning Models,” Alexandria Engineering Journal, vol. 80, pp. 110–121, Oct. 2023. doi:10.1016/j.aej.2023.08.038

[12] M. S. Jahan and M. Oussalah, “A systematic review of hate speech automatic detection using Natural Language Processing,” Neurocomputing, vol. 546, p. 126232, Aug. 2023. doi:10.1016/j.neucom.2023.12623

[13] F. R. S. Nascimento, G. D. C. Cavalcanti, and M. Da Costa- Abreu, “Unintended bias evaluation: An analysis of hate speech detection and gender bias

mitigation on social media using Ensemble Learning," Expert Systems with Applications, vol. 201, p. 117032, Sep. 2022. doi:10.1016/j.eswa.2022.117032

Appendix A

Course Outcomes, Complex Engineering Problems (EP) and Complex Engineering Activities (EA) Addressing

Title: A STUDY OF HATE SPEECH DETECTION IN ONLINE FORUMS USING NLP

Student ID: 203-15-3877, 203-15-3857

CO Description for FYDP

CO	CO Descriptions	PO
Phase -I		
CO1	Integrate recently gained and previously acquired knowledge to identify a real life complex engineering problem for the Final Year Design Project (FYDP)	**PO1**
CO2	Analyze different aspects of the goals in designing a solution for this FYDP	**PO2**
CO3	Explore diverse problem domains through a literature review, delineate the issues, and establish these goals for the FYDP	**PO4**
CO4	Perform economic evaluation and cost estimation and employ suitable project management procedures throughout the development life cycle of the FYDP	**PO11**
Phase -II		
CO5	Design and develop technical solutions and system components or processes that meet specified requirements, ensuring compliance with public health and safety standards, as well as considering cultural, socioeconomic, and environmental factors in this FYDP	**PO3**
CO6	Choose and apply appropriate methodologies, resources, and contemporary engineering and IT technologies to address complex engineering processes, encompassing prediction and modeling, while adhering to relevant constraints in this FYDP	**PO5**
CO7	Analyze societal, health, safety, legal, and cultural considerations, along with associated responsibilities, in the context of professional engineering practice and the resolution of this problem, employing logical reasoning guided by contextual understanding.	**PO6**
CO8	Comprehend and evaluate the enduring sustainability and impact of professional engineering endeavors in addressing intricate engineering challenges within social and environmental frameworks.	**PO7**
CO9	Implement ethical principles and adhere to professional standards and norms in this FYDP	**PO8**
CO10	Capable of operating proficiently both individually and as a team member or leader across diverse teams and interdisciplinary settings in this FYDP.	**PO9**

CO11	Proficiently communicate with the engineering community and broader society regarding complex engineering endeavors, including the ability to comprehend and generate comprehensive reports and design documentation, as well as provide and receive clear instructions throughout this FYDP.	**PO10**
CO12	Acknowledge the importance of self-directed and life-long learning within the evolving landscape of technology and possess the readiness and capability to engage in lifelong learning endeavors.	**PO12**

Addressing CO (1 to 8), Knowledge Profile (K), Attainment of Complex Engineering Problems (EP), and Attainment of Complex Engineering Activities (EA)

Addressing CO (1 to 8), Knowledge Profile (K), Attainment of Complex Engineering Problems (EP):

SN	EP Definition	Attainment	CO	Justification (with Knowledge Profile)	References
1.	**EP1: Depth of Knowledge required**	Yes	CO1 CO2 and CO3	Our thesis is about detecting hate speech from the dataset that we have collected from various online forums and in the methodology section, we have described the process in the methodology section which meets the requirements of engineering fundamentals (K3) and specialist knowledge in deep learning (K4).	**Page no:** [7-9] **Section:** [3.1, 3.2]
				We at first preprocessed the data, extracted the features and tried to predict the model as well which covers the aspects of engineering design (K5) and engineering practice(K6).	**Page no:** [12] **Section:** [3.2]

				We reviewed the existing literature on hate speech detection to inform our methodology (K8).	**Page no:** [5] **Section:** [2.2, 2.3]
2.	**EP2: Range of Conflicting Requirements**	Yes	CO2, and CO7	This project addresses EP-2 by recognizing the hurdles in hate speech detection, including limitations of traditional methods and the complexities of integrating transfer learning and deep CNNs.	**Page no:** [32-34] **Section:** [7.3]
3.	**EP3: Depth of analysis required**	Yes	CO2, and CO6	In this thesis, we applied deep learning techniques to the RNN, LSTM, CNN model and analyzed the experimental results.	**Page no:** [23-26] **Section:** [5.2, 5.3]
4.	**EP4: Familiarity of Issues**	Yes	CO8	As the thesis is about detecting hate speech, there are similar theories regarding this topic that will have a great impact on social life.	**Page no:** [27-30] **Section:** [6.1, 6.2]
5.	**EP5: Extends of application codes**	No	CO3	N/A	N/A

6.	**EP6: Extends of stakeholders involved and conflicting requirements**	Yes	CO8	We annotated our dataset from a group of people and tried to get knowledge about the implementation of our work from stakeholders.	**Page no:** [15-16] **Section:** [3.4]
7.	**EP7: Interdependence**	Yes	CO5	Our thesis has some interdependence like data cleaning, data preprocessing, model training which gives the surety of EP7. Which required some tools and sub tools to implement.	**Page no:** [12] **Section:** [3.3]

Addressing CO11 with Complex Engineering Activities (EA) [Some or all the following]:

SN	EA Definition	Attainment	CO	Justification	References
1.	**EA1: Range of resources**	Yes	CO11	Our thesis has the resources like deep learning models, data annotations which ensures the contribution with the help of RNN.	**Page no:** [10] **Section:** [3.3]
2.	**EA2: Level of interaction**	No		N/A	N/A

3.	**EA3: Innovation**	Yes		The project demonstrates innovation by introducing a completely new dataset specifically tailored for hate speech detection. This framework incorporates a unique deep learning model, contributing to advancements in hate speech detection practices.	**Page no:** [15-16] **Section:** [4.2, 4.3]
4.	**EA4: Consequences for society and the environment**	Yes		These technologies identify and eliminate bad information and potentially discriminatory language, they are crucial for safeguarding online communities. These technologies contribute to safer internet experiences for individuals worldwide by automatically reporting hate speech and fostering more inclusive and polite online discourse.	**Page no:** [28] **Section:** [6.1, 6.2]
5.	**EA-5: Familiarity**	Yes		This thesis is just about a small examination of how accurately a machine can detect hate speech from a random dataset which shows the accessibility of that particular model used in the thesis.	**Page no:** [4-5] **Section:** [2.1, 2.3]

Addressing CO (4, 9, 10, and 12):

SN	COs	Attainment	Justification	References
1	CO4	Yes	To address CO4, the thesis has effective management and financial works like the estimation of the utilization during the completion process.	**Page no:** [13-14] **Section:** [3.4]
2	CO9	Yes	As we collected data from social media, it didn't bother any individuals' privacy and it is all about ensuring social awareness.	**Page no:** [29-30] **Section:** [6.3]
3	CO10	No	N/A	N/A
4	CO12	Yes	The main intention of this thesis is to make a machine more accurate in	**Page no:** [15-17]

			detecting hate speech from any random dataset, to analyze the methodology more accurately and to add new methods if needed which fulfills the requirement of (CO12).	**Section:** [4.1]

Hate speech detection

ORIGINALITY REPORT

6%	3%	0%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	5%
2	aclanthology.org Internet Source	1%
3	Submitted to University of Hertfordshire Student Paper	<1%
4	Submitted to American Public University System Student Paper	<1%
5	ikee.lib.auth.gr Internet Source	<1%
6	www.semanticscholar.org Internet Source	<1%
7	e-journal.unair.ac.id Internet Source	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		