A Major Project Work – Phase I on

Comprehensive System for Cyberbullying Detection and Topic Analysis on Twitter

Submitted in partial fulfilment of the requirements of the award of the Bachelor of Technology

in

Department of Computer Science and Engineering (Artificial Intelligence and Machine Learning)

by

A. Priya Krishna	**21241A66D1**
B. Chamundeshwari	**21241A66D9**
P. Nandini	**22245A6617**

Under the Esteemed guidance of

Ms. B. Kiranmai

Assistant Professor

Department of Computer Science and Engineering
(Artificial Intelligence and Machine Learning)

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Approved by AICTE, Autonomous under JNTUH, Hyderabad) Bachupally, Kukatpally, Hyderabad-500090

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

Hyderabad-50009

CERTIFICATE

This is to certify that the major project entitled "**Comprehensive System for Cyberbullying Detection and Topic Analysis on Twitter**" is submitted by **A.Priya Krishna(21241A66D1), B.Chamundeshwari (21241A66D9) and P.Nandini (22245A6617)** in partial fulfillment of the award in BACHELOR OF TECHNOLOGY in Computer Science and Engineering (Artificial Intelligence and Machine Learning) during academic year 2024- 2025.

Internal Guide	**Head of the Department**
Ms. B. Kiranmai	**Dr. G. Karuna**

External Examiner

ACKNOWLEDGEMENT

There are many people who helped us directly and indirectly to complete our project successfully. We would like to take this opportunity to thank one and all. First, we would like to express our deep gratitude towards our internal guide **Ms. B. Kiranmai, Assistant Professor,** Department of Computer Science and Engineering (Artificial Intelligence and Machine Learning), for his/her support in the completion of our dissertation. We wish to express our sincere thanks to **Dr. P. Sanjeeva,** Associate Professor, Project Coordinator, **P.Deepthi,** Assistant Professor, Project Coordinator for timely suggestion**, Dr. G. Karuna,** Head of the Department**,** and to our principal, **Dr. J. PRAVEEN,** for providing the facilities to complete the dissertation. We would like to thank all our faculty and friends for their help and constructive criticism during the project period. Finally, we are very much indebted to our parents for their moral support and encouragement to achieve goals.

A. Priya Krishna(21241A66D1)

B. Chamundeshwari (21241A66D9)

P. Nandini (22245A6617)

DECLARATION

We hereby declare that the major project titled **“A Comprehensive System for Cyberbullying Detection and Topic Analysis on Twitter”** is the work done during the period from **2th August 2024 to 29th November 2024** and is submitted in the partial fulfilment of the requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering (Artificial Intelligence and Machine Learning) from Gokaraju Rangaraju Institute of Engineering and Technology (Autonomous under Jawaharlal Nehru Technology University, Hyderabad). The results embodied in this project have not been submitted to any other University or Institution for the award of any degree or diploma.

A. Priya Krishna(21241A66D1)
B. Chamundeshwari (21241A66D9)
P. Nandini (22245A6617)

ABSTRACT

As social media platforms like Twitter continue playing a significant role in public conversation, hate speech and cyberbullying are growing in frequency. As a way to stop it, significant monitoring and analysis methods are now required. By developing an AI-based system trained in identifying cyberbullying and tracking popular themes from the relevant Twitter feed using sophisticated natural language processing and deep learning.Using a custom Selenium WebDriver, this system records information in real-time by continuously scrolling through all of the tweets that include the keywords or hashtags specified for the collection. The CNN and LSTM hybrid architecture then classifies the information obtained into three categories: neutral, hate speech, and cyberbullying. The system also does sentiment analysis and topic modelling to gain a deeper understanding of popular subjects and public opinion. These conclusions are shown on an interactive Streamlit dashboard with characteristics for quickly exploring trends, categorized tweets, and visualization. Researchers, legislators, and other groups who handle online harassment and maintain track of Twitter trends can discover the insights needed to understand the scope of online harassment and respond proactively.

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

LSTM	Long Short Term Memory
CNN	Convolution Neural Network
VADER	Valence Aware Dictionary and Sentiment Reasoner
NLP	Natural Language Processing
NLTK	Natural Language ToolKit
LDA	Latent Dirichlet Allocation

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The following section describes the significance of this Twitter-based AI system which plays a crucial role in detecting cyberbullying and hate speech in real-time while preserving the flow of online conversations and uncovering trending topics.

1.1. Introduction to project work

Social media platforms have transformed the way individuals communicate, connect, and share information. However this rapid expansion has also given rise to significant challenges such as the proliferation of hate speech and cyberbullying. These issues not only affect individual well-being but also pose a threat to societal harmony. From the years, researchers and technologists have explored various methods to address these concerns, leveraging natural language processing (NLP) and artificial intelligence (AI) to identify harmful content online. Earlier few systems were used for monitoring online abuse which often relied on keyword matching and rule-based algorithms which were effective but limited in capturing the contextual nuances of modern-day language such as sarcasm or implicit hate speech.

Advancements in machine learning and deep learning have ushered progress in a new era of hate speech detection systems. Models such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have been widely adopted for text classification tasks. These models are capable of learning complex patterns in textual data and enabling them to identify fine differences between benign and harmful content. For instance, hybrid architectures combining CNNs for feature extraction and LSTMs for capturing temporal dependencies have shown promise in detecting contextually intricate cases of cyberbullying. This progression has enabled the development of systems which not only detect harmful language but also analyze sentiment and predict the emotional nuances of the conversations. Combating cyberbullying projects have also looked into user reporting and content moderation systems.

For instance, some have added community moderation functionality to their platforms, which allows for content flagged by users to be examined by certain moderators. While these methods are efficient, they are also very slow and can hardly be applied to platforms with millions of users. Machine-based detection is an answer to this problem as it allows for real-time feedback on any large volume of information, thanks to its ability to analyze nearly unlimited amount of information at a given time. Solutions like the google's perspective api and the facebook's ai significance of the moderation systems are that aimed at tackling the issue of toxic content. Such systems, however, are usually criticised for their negative lack of transparency and biased approaches and emphasize the necessity for a better structure.

Within the educational and scientific fields, there have been a number of attempts to build publicly available datasets and benchmarks for hatred and emotions analysis. For instance, the Hate Speech And Offensive Language dataset and Cyberbullying Detection dataset come with a range of labelled comments and tweets. These resources serve the purpose of model training as well as performance testing of many machine-learning models. Also, other researchers have attempted to use unsupervised learning techniques, such as topic modelling, to expose latent aspects of such discussions, which in turn helps shed more light on the issues of social networks.

In recent years, systems intended for the monitoring and enforcement of online behavior have also gained popularity. They monitor hashtag and keyword activity to understand the existing public and predict the possible ongoing or upcoming trends. Losing sight this time, sentiment analysis becomes a key factor because it gives an understanding of how people feel about certain matters or occasions. LDA, BERT, and similar methods are typical implementations for the purpose of topic modelling from the unstructured textual information. These technologies can be fused together in a single design to not only perform the basic functions of the system, which allows to tackle the problem of online harassment, but also utilize the data for the purposes of business intelligence.

The combination of detecting hate speech and identifying sentiment as well as monitoring trends helps in appreciating the effects of social media interactions to a very huge extent. It helps in establishing the presence of deleterious interactions and offers a larger context of the societal problems. This is where researcher and policy maker's interest focusses as well as for the organizations that deal with online bullying and advocating for the safe digital environment. This is made possible through incorporation of modern day natural language processing and deep learning as a tool in cyberspace.

1.2 Objective of the Project

The main aim of this project is to create an artificial intelligence based tool for on-the-spot monitoring and assessment of harmful content, concentrating mainly on issues of cyberbullying and hate speech on the Twitter platform. The system employs a customized Selenium Web Driver to scrape tweets from target accounts, hashtags or topics when the need arises. It differentiates hate/troll and cyberbully tweets from neutral ones by employing a hybrid system that combines Convolutional Neural Networks in the text processing unit and Long Short-Term Memories in the dependency capturing unit. In addition, the system also incorporates opinion mining and topics detection to analyze the prevailing emotions, trends and topics of interest. The explained features are provided on a user-friendly dashboard created with Streamlit, which allows easy visualization of sentiment and topics among users. This system is designed to help researchers, organizations, and even governments in monitoring and evaluating negative patterns and public opinion respectively and in countering the growing threats of cyberbullying and hate attacks.

1.3 Methodology Adopted

In this section the following information is presented:

- Data extraction using Selenium WebDriver.
- Text Processing with CNN.
- Capturing text dependencies with LSTM
- Sentiment Analysis with VADER
- Topic Analysis with LDA

Data extraction using Selenium WebDriver

Figure 1.1 In this project, Selenium's WebDriver is used to automate the real-time collection of tweets from a predefined set of Twitter users, hashtags or topics. The WebDriver then scrapes the Twitter feeds by continuously scrolling the timeline to fetch all the tweets meeting the given specifications. This method of data collection is advantageous in that it does not involve human efforts enabling it to collect data in large quantities in a very short period of time. Users are allowed to narrow down the data gathered by selecting certain words or account names and this makes the process of gathering data very active and less passive. This particular means of data collection is very useful in tracking incidents of cyberbullying and hate speech on Twitter, allowing analysts to intervene when necessary.

Selenium WebDriver has the capability of automating the whole process of collecting tweets thus can interoperate with other parts of the system with ease. The WebDriver also guarantees that the tweets that are collected are in congruence with the prevailing conversation trends by performing actions such as scrolling, clicking, and navigating the website as a user would. This is especially important when considering how harmful content continues to spread and it is desirable to be able to collect and analyze such data in real-time.

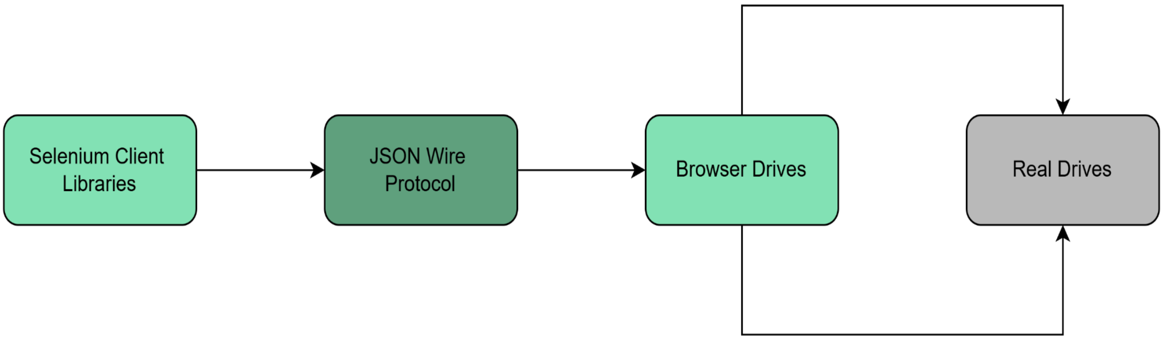


Figure 1.1: Data extraction using Selenium WebDriver

Selenium WebDriver has the capability of automating the whole process of collecting tweets thus can interoperate with other parts of the system with ease. The WebDriver also guarantees that the tweets that are collected are in congruence with the prevailing conversation trends by performing actions such as scrolling, clicking, and navigating the website as a user would. This is especially important when considering how harmful content continues to spread and it is desirable to be able to collect and analyze such data in real-time. Furthermore, the feature to set narrow collection parameters ensures that only the relevant data is presented for analysis, which in turn improves the efficiency of the whole system.

Text Processing with CNN

Figure 1.2 In this project, the Convolutional Neural Networks (CNNs) are exploited to analyze the textual data included in tweets and thus, utilize effective means for feature vector generation. Unlike simpler approaches, such as the bag of words model, CNNs appear to work better with the data by recognizing the members which make up the text structure within it, for example, certain keywords and phrases that promote hate or may be seen as cyberbullying. The design of the model incorporates several convolutional layers inclined at an angle where the crucial elements of the structure are visible. This ensures that the correct classification of tweets into the various targeted groups is achieved. Given its architecture, CNN's are fast and efficient on computing, making it suitable for this application which requires real time analysis.

The use of CNNs in text processing further improves the system's capabilities in identifying complex traces of offensive content, which traditional techniques are likely to miss. With the hierarchies in patterns of text created by CNNs, the model is able to comprehend some aspects of a language, such as, context, tone and intention. This makes it easier to categorize tweets as neutral, hate speech, or cyberbullying.

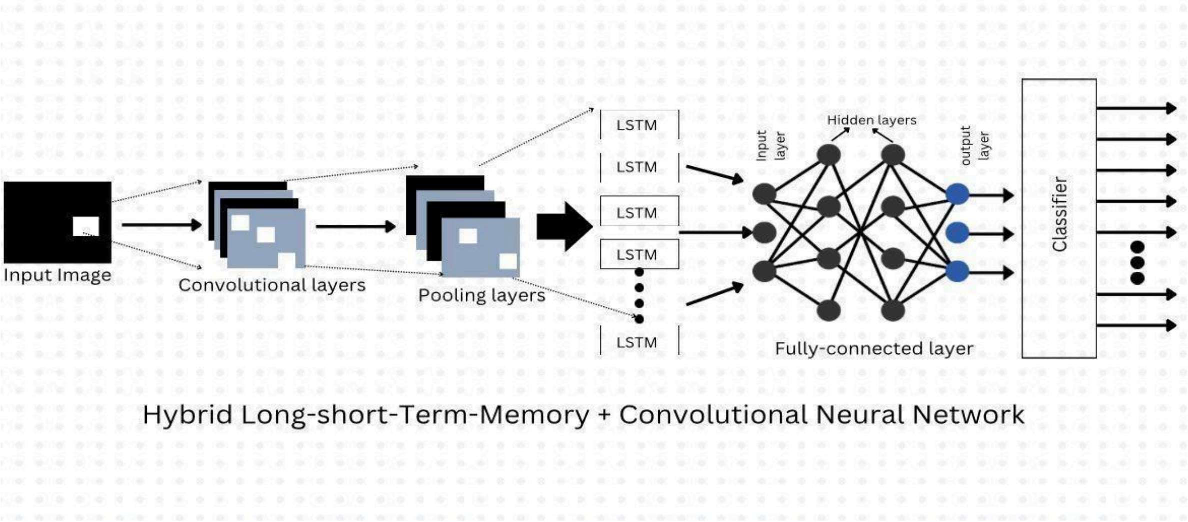


Figure 1.2: Text Processing with CNN

When used for analysing the twitter data which is s rather unstructured form of communication where tweet sentence structures vary, CNNs prove to be very resourceful as they can be used for the project objective of monitoring and analysing online abuse in real time.

Capturing text dependencies with LSTM

Figure1.3 LSTM networks are applied to enable recognition of the relationship of the words present in a tweet to the occurrence of particular words in a given context. This is in contrast to nural networks which are typically designed to work with non-sequential data. LSTMs on the contrary are built to work with chronological data such as sentences where word order is very determinant of the meaning of the words.This is critical especially in understanding the language used in such concise messages as tweets, where sarcasm, and tone or any other contextual aspect could completely alter the meaning of a statement. In addition, because LSTMs explore how words relate to one another over time, they also reduce systems error

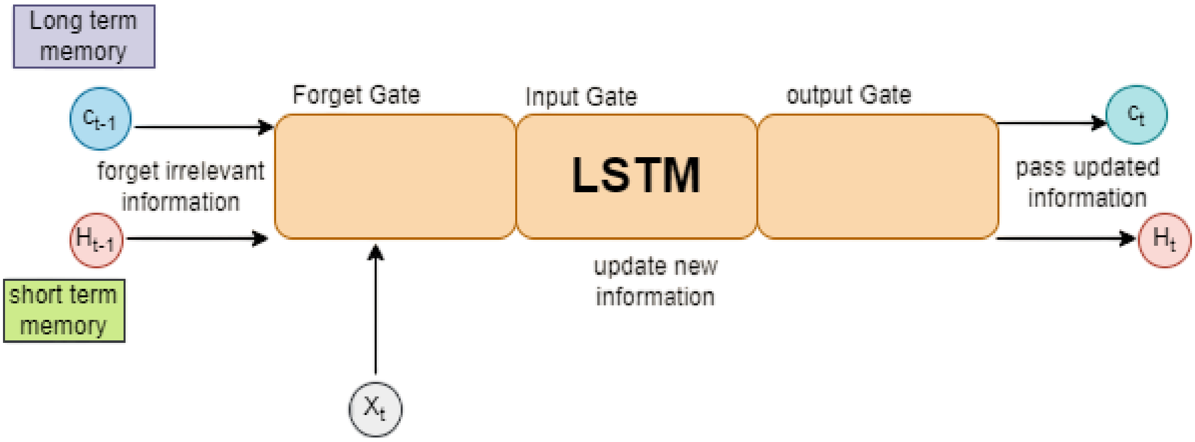


Figure 1.3: Capturing text dependencies with LSTM

in tweet classification.The integration of CNNs with LSTMs allows the model to gain non-linear sequential aspects to the hidden representation of the individual words, overriding the limitation of examining only the words in isolation. Since this combines feature maps (CNN) and sequence networks (LSTM), a lot of comprehension of text is possible which is important in identifying subtle cases of cyberbullying or hate speech. The LSTM deep learning model can even discriminate 'safe' tweets and 'toxic' ones, which is critical for detection and classification responses, even though the aggression is couched within other innocuous positive messaging.

Sentiment Analysis with VADER

Figure 1.4 This project features sentiment analysis as a significant aspect, in which VADER (Valence Aware Dictionary and sEntiment Reasoner) has been used to understand the emotional tone of tweets. VADER is a sentiment analysis engine based on a dictionary and a set of rules. It is primarily focused on extracting a sentiment from social media texts. It has been designed to classify the sentiment in tweets as positive, negative or neutral based on sentiment lexicons. Since emoticons, slang and intentional capitalization are all part and parcel of Twitter language, VADER is primarily used for sentiment analysis of Twitter posts. This assists in determining the underlying emotional magnitude of the context within which cyberbullying and hate speech occur.

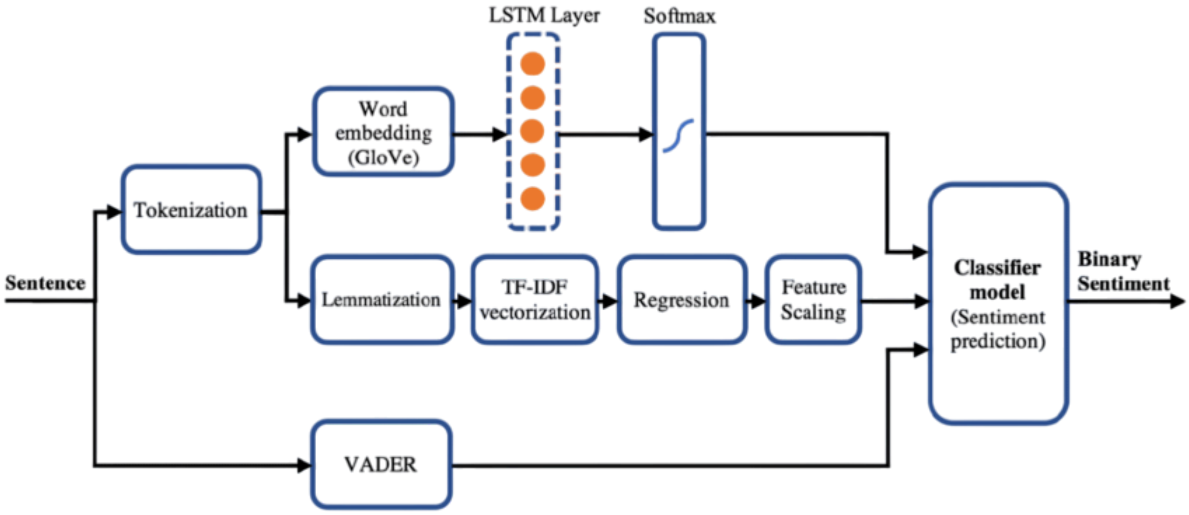


Figure 1.4: Sentiment Analysis with VADER

By applying VADER, the system can classify tweets not just by content but by the emotional tone they convey. This sentiment analysis provides valuable insights into how certain topics or tweets are perceived by the public, aiding in the identification of not only harmful content but also the general mood of online discussions. The ability to gauge sentiment in real time allows for more nuanced insights into public reactions, which is especially important for tracking the emotional response to issues like cyberbullying. This feature complements the broader goal of understanding online behavior and addressing harmful interactions effectively.

Topic Analysis with LDA

Figure 1.5 shows the architecture of LDA.This project also encompasses topic modeling and the uncovering of latent topics in the tweet dataset using Latent Dirichlet Allocation (LDA). LDA is a well known non-guided machine learning method that helps in finding certain clusters of words which keep on occurring together in a large body of text enabling the system to identify the main subject or topics being talked about in the tweets. The approach analyzes the topics of conversation and understands what topics are trending or receiving the highest attention, which is helpful in qualitative analysis of public sentiment and other ideas pertaining to the topics of interest practice, cyberbully and hate speech.

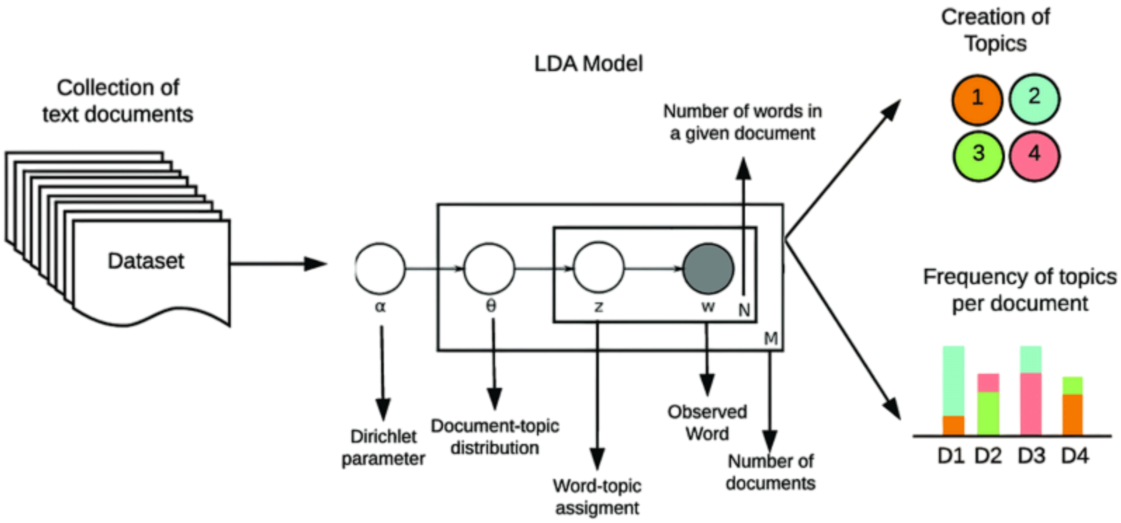


Figure 1.5: Topic Analysis with LDA

In this manner, the algorithm gets the opportunity to receive a fresh breeze in moments where there is no presence of information through tweets by identifying the main topics being discussed. This step contributes to the identification of trends in the data including, for instance, the presence of hate speech or cyberbullying in relation to certain topics or discussions. Topic modelling is also useful for the understanding of opinion meaning, since it enables to tie opinion about an object with discussion concerning that object. But this trend and topic monitoring is also useful in helping one understand the status of online engagement behaviorally and aids in decision making when actioning on derelict contents online.

1.4 Block Diagram

Figure 1.6 is the block diagram of the text analytics pipeline workflow on specific focus for tweets. The process starts with Data Collection and Extraction where tweets are derived from platforms using Twitter APIs or web scraping. It continues to process through CNN (Convolutional Neural Network) and LSTM (Long Short-Term Memory) models for natural language processing tasks. Then, the collected data is subjected to Classification of Tweets wherein categories or labels are assigned (e.g., spam vs. non-spam or specific topics). Then Sentiment Analysis is carried out to find out the emotional tone, whether this is positive, negative, or neutral. Thirdly, Topic Modelling focuses on finding underlying themes or subjects being discussed by the tweets. Finally, the results are put through visualization in a Dashboard so users can obtain insights from these graphical representations.

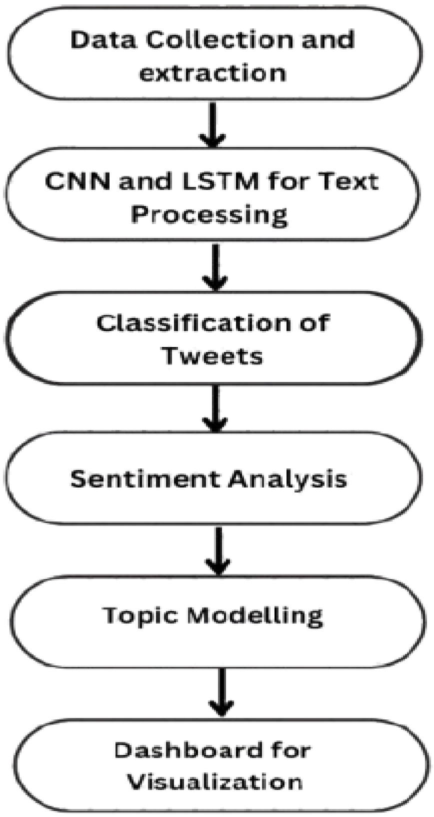

Data Collection and extraction
CNN and LSTM for Text Processing
Classification of Tweets
Sentiment Analysis
Topic Modelling
Dashboard for Visualization

Figure 1.6: Block Diagram

1.5 Organisation of the Report

Introduction

This section presents the goal, the problem statement, the approach employed and contains the architecture of the project. The architecture diagram indicates the flow of the system and highlights important phases such as data collection, processing, classification and then data visualization. It demonstrates the whole process of tweeting extraction, classification of the tweets using CNN and LSTM, performing sentiment analysis and topic analysis and finally presenting the results using Streamlit.

Literature Survey

Existing literature spanning research articles, journals, and conference papers is researched in order to comprehend the current techniques employed for detection of cyberbullying and classification of hate speech. This section identifies the merits, demerits and advantages of most related works in order to appreciate the system being proposed.

Proposed Method

This section provides the goal and the problem statement of the project which is aimed at collecting data with the help of Selenium WebDriver and classifying it using CNN and LSTM. It outlines the processes of sentiment and topic analysis, the architecture diagram, specs for software and hardware as well as functional and non-functional aspects of the system. As for the design of the system, UML diagrams are incorporated.

Results and Description

The section describes the dataset, the results achieved, and the processes utilised in achieving said results. It includes the analysis of tweets classifications, sentiments, and topics modelling together with model performance and the drawbacks that necessitate the improvement of the model in question.

Conclusion and Future Enhancements

There is a brief overview of the project highlights, sectional modules and methodologies employed in the course of the study. Also included in this section are the possible future improvements such as inclusion of more networks and better classifiers.

Appendices

In this section, we focus on the accompanying sample code and its further implementation details for which we provide adequate references.

1.5 References

All articles, journals or internet resources consulted in preparation of the project are properly documented so that the project stands on its own and gives credits for existing research

CHAPTER 2

LITERATURE SURVEY

2.1 Summary of Manuscripts

Mitushi Raj and team [1], proposed an Application to Detect Cyberbullying Using Machine Learning and Deep Learning Techniques. The model presents a deep learning-based cyberbullying detection system which can examine social media posts in English, Hindi, and Hinglish. To specifically detect posts that contain cyberbullying, the system uses a CNN-BiLSTM model that has been built on preprocessed datasets. The dataset has been generated for model training by a combination of data transformation, reduction, and cleaning techniques. With 98 percent accuracy, the CNN-BiLSTM model performed better than the others. Social media platforms may utilize this technique to automatically monitor hazardous information in real time. The study shows how important deep learning is to identifying cross-lingual cyberbullying, and it could possibly be generalized to multimodal data like pictures and videos.

Aditya Desai and team [2], proposed a model on Cyber Bullying Detection on Social Media using Machine learning. Using the BERT model and five key characteristics-sentimental, sarcastic, syntactic, semantic, and social media. This research suggests a semi-supervised method for detecting cyberbullying on social media. The model exceeded classic machine learning models like SVM and Naive Bayes, achieving 91.90% accuracy in sentiment analysis. By including various kinds of linguistic and social characteristics, the suggested methodology goes beyond basic sentiment analysis and improves its capacity to recognize a wide range of bullying actions. Having many restrictions, the BERT model's outstanding performance shows its effectiveness. The dependence on huge, diverse datasets and opaque decision-making processes may have an effect on its implementation in practice. To further improve detection accuracy and use, future studies could look into growing the dataset, adding more features, and creating hybrid models.

Aljwharah Alabdulwahab and team [3], proposes a model for detecting cyberbullying on social media platforms using a variety of machine learning techniques and deep learning. The primary objective of their research was to utilise various kinds of NLP and machine learning methods, such as KNN, SVM, and DL, the goal is to identify incidents of cyberbullying. According to the deep learning there are particularly 6-layer Convolutional Neural Network models performed higher than conventional methods, with an accuracy of 96% as compared to 90% for KNN and 92% for SVM. This approach shows the importance for efficient NLP for avoidance of cyberbullying by identifying bullying behavior in tweets using feature extraction techniques. While the performance accuracy of this deep learning model is good, it is computationally expensive and requires a lot of huge datasets. Thus, the study recommended implementing the multi-channel design, increasing algorithm efficiency, and growing dataset sizes to boost performance and extend this framework to multi-language datasets.

Jafri Sayeedaaliza Abutorab and team [4], proposed the model for machine learning-based social media cyberbullying detection discusses the application of natural language processing (NLP) methods and machine learning algorithms to detect harmful content on social media sites such as Twitter. This research was conducted to identify instances of cyberbullying on social media, using a specific focus on Twitter as the statistical target. Naïve Bayes, Deep Neural Networks, and Support Vector Machines were utilized to implement the machine learning approach. Optical Character Recognition is utilized for image-based detection, as Natural Language Processing is used for risky text detection. Based on reports, BERT has achieved a 91.90% accuracy rate for sentiment analysis as it relates to detecting cases of cyberbullying. It was made very clear that the automation method will reduce the quantity of manual monitoring that is performed for recognizing bullying content in real time. Live tweets are handled by a Flask-developed web application, which provides an automatic and extremely accurate detection technique.

John Hani and team [5], proposed a model on Social Media Cyberbullying Detection using machine Learning. It provides a machine learning method to detect instances of online cyberbullying using preprocessing, sentiment analysis using N-grams, TFIDF feature extraction, and further classification using SVM and neural networks for classification. A Kaggle cyberbullying dataset was used for the process of evaluation, and the results suggested that neural networks exceeded the support vector machines (SVMs) with an accuracy of 92.8% and an average F-score of 91.9%, in contrast to SVMs' 90.3% accuracy and 89.8% F-score. The proposed approach outperforms the state of the art, showing the value of combining different methods of machine learning to detect negative online conduct. The model continues to face issues with dataset imbalance and size, and it could not be able to manage non-textual kinds of cyberbullying, which further emphasizes the need for larger datasets and more holistic models in future work.

Daniyar Sultan and team [6], proposed a system on Cyberbullying-related Hate Speech Detection Using Shallow-to-deep Learning. This study examines the accuracy, precision, recall, and F1-score of shallow and deep learning algorithms for detecting cyberbullying on social media. The results show that deep learning models especially BiLSTM across different datasets—perform higher than shallow machine learning models in the detection of cyberbullying. As a result, it displayed strong AUC-ROC values and adaptability to both short and long texts. Also, the research provides a basic metadata architecture for categorizing cyberbullying that addresses the problem of multilingualism, imbalanced data, and fragmented data. For example, while this technique offers scalable and accurate real-time detection methods, it is incapable for handling non-textual cyberbullying and involves a significant processing cost, making it less applicable in areas with limited resources.

RajuKumar and ArunaBhat [7], proposed a machine learning-based models fordetection, control and mitigation ofcyberbullying inonline social media. A detailed examination of machine learning (ML) and deep learning (DL) models for recognizing, handling, and avoiding instances of cyberbullying on online social media (OSM) is offered in this research. Thus, various kinds of cyberbullying and the critical requirement for predictive algorithms in order to protect sensitive individuals are mentioned. This paper's description of crucial methods, including data pretreatment, labeling, and classification; the problem of data imbalance; privacy issues; and real-time processing, is an intriguing aspect. It highlights the importance of ML and DL in identifying objectionable information and provides guidance for scholars working in this area. However, it also notes that there are obstacles, such as the scalability of the model used in the study and the availability of sizable, varied datasets. It makes clear just how essential it is to continue improving these systems in order to better protect people in this growing digital world.

Edla Hareen [8], proposed focuses on detecting cyberbullying on social media platforms using deep learning-based hybrid models. Previous approaches used handcrafted features, which includes the ability to identify swear words, which range widely in bullying styles across several platforms including Wikipedia, Formspring, and Twitter. Class imbalance, platform-specific language, and topic variety are some of the primary problems to be addressed. Results showed that oversampling and transfer learning techniques generated better enhancements, leading to increased precision, recall, and F1 scores. One significant advantage, displayed below, is that it has outperformed standard machine learning models, at an accuracy of up to 92.81%. Also this study showed that models using deep learning perform conventional techniques and can be exported across datasets with minimum retraining. By identifying cyberbullying in a way that is reliable, scalable, and context-aware, the results will contribute to the development of a safer online environment.

Ravindra Chilbule and her team [9], present a paper on hate speech detection on online platforms using machine learning and natural language processing (NLP) techniques. The methods, difficulties, and developments in the area were utilized when comparing deep learning techniques to traditional machine learning models. The main conclusions indicate that decision trees, ensemble methods, transfer learning algorithms, and feature engineering—such as n-grams and word embeddings—are helpful in identifying hate speech patterns. It addresses problems including context sensitivity, class imbalance, and new developments in language and offers solutions like data augmentation and flexible models. While the suggested approaches show highly encouraging results in detecting accuracy increases, they are limited by their dependence on high-quality labeled datasets and their sensitivity to cultural and linguistic variations. Recognizing the urgency to minimize harm and create a safer online environment, the implications of these findings also suggest for an even more robust hate speech recognition system.

Sneha Gajanan Sambare and her team [10], present a paper detection of cyberbullying on social media through approaches in machine learning and natural language processing.The suggested strategy would use datasets with SVM and a random forest for classification, analysing hate speech on Twitter and personal attacks on Wikipedia. The findings indicate that the BoW and TF-IDF models are far more effective than Word2Vec at identifying hate speech on Twitter, with accuracy rates of over 90%. However, due to the subtlety of such remarks, it was more difficult to detect personal assaults. Word2Vec models and multi-layered perceptrons performed well in terms of contextual understanding of the comment. The real-time use of this research to reduce the negative impacts of cyberbullying, such as depression and other mental health problems, is of great importance. Future research should incorporate increasingly complex techniques and take into account how attacks are changing.

Cinare Oguz Aliyeva and Mete Yaganoglu [11], developed a system with Deep learning approach to detect cyberbullying on twitter.Despite Turkey's high rates of cyberbullying, much is known about the severe issue of cyberbullying among Turkish children and adolescents, which is the subject of this study. The study proposes a straightforward but efficient MLP-based deep learning model for identifying cyberbullying in Turkish Twitter tweets by combining textual and social media aspects. With a testing accuracy of 93.2%, a high F1-score, and a brief training period, the model outperformed both conventional machine learning and cutting-edge deep learning models (such as CNN, LSTM, and BERT). The efficiency and generalizability of the model were enhanced by the addition of synthetic data. The focus on a single dataset and language represents one of its drawbacks; future research should incorporate sentiment and user-based attributes and test on a variety of platforms. The complete methodology used in this study greatly advances the identification of cyberbullying in low-resource languages.

S.Logasree and M.Harshini [12], proposes a system for detecting cyberbullying using machine learning and natural language processing. With a 78.5% accuracy rate, the system will be built to use a Naïve Bayes classifier to analyze abuse content in real time. The technique was put into use on a live chat program that provided real-time cyberbullying alerts. Although it is less precise than deep learning models, it is accurate for automation and real-world applications. With such sophisticated methods, its accuracy can be improved. The accuracy can be further enhanced. It is possible to incorporate multimedia content. The system is able to identify intricate roles in cyberbullying and how they interact. The victim's emotional analysis and streaming data processing could make the system more resilient and responsive. Developments of this kind would increase monitoring effectiveness and provide much-needed insight into social media management.

Neelakandan and team [13], proposed Feature Subset Selection with Deep Learning-based Cyberbullying Detection and Classification (FSSDL-CBDC) method that uses the BCO algorithm in the feature selection process and an SSA-tuned DBN in the classification of objects. At 95% on benchmark datasets, the proposed approach demonstrated better performance when compared to more conventional methods like NB, SVM, and ANN-DRL. Robust hyperparameter optimization, effective feature subset selection, and effective classification performance are its primary benefits. The method is computationally expensive, nevertheless, and may need refinement when dealing with outlier detection or real-time scalability problems for high-dimensional data streams. Further enhancements could also include expanding the range of applications, such as intrusion detection, and other jobs that need for quick data analysis.

Dong-Hwi Kim and team [14], proposed a Hybrid Deep Learning Emotion Classification System that utilizes multimodal data, including text, audio, and biodata, to solve challenging emotion classification issues, especially for languages with complex NLP (natural language processing) characteristics like Korean. HDECS shows a significant improvement in performance over the base KLUE/roBERTa model by integrating CNN and LSTM models applied to biodata and audio analysis as well as enriched textual input with BERT. This results in a higher accuracy of 0.09, a higher F1 score of 0.11, and an especially low loss of 0.5 less than the baseline value of 0.35. Once more, compared to previous implementations, the architecture is better able to handle multichannel data and multimodal problems. While it requires a lot of processing power, this also improves accuracy and resilience for classification purposes. Other multimodal inputs, such pictures or facial expressions, have not been fully investigated. It is recommended that future research focus on modalities and develop ensemble approaches for wider applicability across languages and contexts.

Md Saroar Jahan and Mourad Oussalah [15], proposed a move away from traditional machine learning techniques like SVM and TF-IDF to highly efficient deep learning architectures like CNN, RNN, and BERT that use word embeddings like Word2Vec, GloVe, and FastText is the backdrop for this critical systematic review, which examines the advancements made in hate speech (HS) detection with NLP and deep learning. Although these successes, there are still issues: there aren't many large, high-quality datasets; there aren't many architecture comparison studies; and there isn't much study on tasks that aren't in English. The research paper also examines emerging multimodal detection methods that use on visual, aural, and textual data. The research claims that it has already observed positive results for models like VisualBERT and XML-RoBERTa. Among the advantages are the demonstrated effectiveness of a CNN + LSTM combined deep learning model and the application of contextual embeddings in HS detection, like BERT. The lack of appropriate resources for non-English languages, domain gaps in the multimodal data, and dataset constraints are limitations, nevertheless. To advance the field of study, such resources, datasets, and additional research into multimodal and multilingual systems are needed.

Madahana and team [16], proposed how hate speech detection and sentiment analysis are moving from traditional machine learning models, like SVM and NB, to advanced deep architectures, including CNNs, RNNs, and especially models that rely on transformers, such as BERT and GPT. The paper also raises the following key challenges: the paucity of labeled datasets, complexity based on language used, data imbalance, and the inherent difficulty of detecting implicit hate speech-sarcasm and stereotypes. While transformer models are great in contextual understanding and state-of-the-art achievements, there are certain constraints such as having inadequate multilingual datasets, a subjective approach to interpreting hate speech, and high costs involved with annotation. Conclusions End The study insists that transfer learning, multilingual resources, and their extension to related NLP tasks like cyberbullying and fake news detection be thus applied to enhance the impact of the application on society and work through possible obstacles.

Donia Gamal and team [17], proposed an intelligent multilingual cyber-hate detection in social networks, with taxonomy, methods, datasets, and challenges. It reviews machine learning and lexicon-based approaches in the detection of cyberhate across multiple languages, where the main significant research gap is related to non-English languages, especially the Arabic language. This paper introduces algorithms which achieve high accuracy ranging from 60% to 94%, where CNN and SVM present the most robust performance in cyber-hate detection. The growing problem of cyber-hate is mentioned on social media platforms, and the urgency for effective multilingual detection techniques is underlined. In addition, the paper also defines a major constraint as it cannot outline annotated datasets for less studied languages and requires greater resources that might be needed for strong detection systems to handle online hate speech diversity in linguistic and cultural expressions.

Douglas C. Youvan [18], have introduced challenges and advancements in the detection of subtle hate speech on social media using AI, shedding light on the necessity of deeper approaches such as NLP models and multimodal analysis in enhancing contextual understanding. The paper shall especially try to focus efforts on improving the accuracy of detection by trying to reduce biases in AI algorithms and increase sensitivity to nuanced cues in text. The research focus towards modulating inappropriate content without infringing free speech ideas tries to create less harmful online environments and hence underlines the need for more robust datasets and continuous learning mechanisms to sustain changing language as well as societal norms. The study has brought to light the critical role of AI in moderating social media but with regard to cultural sensibility and potential biases. In this manner, besides the ethical concern, there would also be a requirement to conduct more research and develop interdisciplinary collaboration towards creating AI systems that would be effective moderators of subtle hate speech.

Edemealem Desalegn Kingawa and team [19], proposed on techniques used to detect hate speech with a focus on quality and performance specifically for the under-researched Amharic language in the domain. It demonstrates the urgency in identifying hate speech and misinformation as early as possible to contain their influence. It discusses collection of relevant posts and comments that are annotated, and machine learning algorithms Naive Bayes and Random Forest are applied on it in this research. The word2vec embedding-based model achieves an accuracy of 79.83%. Challenging is the lack of benchmark datasets and standardized NLP tools for the Amharic language, and neither has precise guidelines for data annotations. The difference hinders the advancement of developing a detection system whose efficiency comes across as effective for non-English languages, largely in the light of an increasing use of hate speech and misinformation or false information within African countries. The paper thus deals with developing collaboration and working on large data collection for making detection techniques more powerful.

Md. Tarek Hasan and team [20], proposed the DL-based methods for cyberbullying detection, which underlines the increasing importance of online harassment identification as a way to fight its effects on the psyches and emotions of individuals. Beyond methods for data representation and model architectures, it identifies the main challenges related to DL techniques, such as the requirement for large datasets and sufficient computing power. It highlights the cultural and linguistic challenges in cyberbullying across diverse populations. Future research includes the development of detection systems that are multilingual and multimedia, using word embeddings especially designed for this task, and the infusion of mental health insights into models. Although the DL methods shine bright, there are several limitations to them, mainly including that they require vast, clean data, and the predictions from DLs are non-interpretable. However, developments in DL may herald more efficient and complete cyberbullying detection mechanisms in the future.

Vijayakumar and team [21], proposed challenges and advances in detecting cyberbullying on multilingual and multimodal platforms-including text, images, and videos. This is something that has become necessary with the recent spate of content across languages and media outlets in this social media age. The work highlights the importance of designing approaches to deep learning methods that automatically analyze cyberbullying in multilingual and multimodal data as well as present algorithms that can keep pace with new linguistics and neologisms-in the form of slang and abbreviations. The core problems remain about data gathering, pre-processing, and fusion, and the lack of resources for models trained across languages. Despite all these barriers, it focuses on the potential of multimodal and multilingual approaches to increase early detection and prevention to benefit users, parents, and social organizations. The capability of deep learning in overcoming such complex data is an advantage in this domain; however, the data storage, feature extraction, and the frequent need for model training updates are drawbacks.

Zainab Mansur and team [22], proposes developing low-dimensional, modality-agnostic algorithms that are capable of detecting cyberbullying; the modality-agnostic nature comes from the fact that the content from social media can be textual, pictorial, or even video. The paper focuses on applying machine learning classifiers and deep learning models on user-generated content across languages. Of course, it focuses on the important role of hyperparameter tuning for optimal results in deep neural networks. In this regard, the study addresses the very critical and international issue of cyberbullying by providing automated detection systems that advance online user safety. Furthermore, the paper suggests that the scope of research in multi-lingual-based detection of cyberbullying is relatively limited and that there exists a serious deficiency in the provision of comprehensive solutions for non-English languages. The approach presented here comes along with several benefits in terms of improved scalability and user safety across languages, though it also calls for several drawbacks with several challenges in handling diverse data types and languages, requiring continuous updates and fine-tuning for accuracy.

Miriam Di Lisio and team [23], proposed hate speech and algorithmic bias in verbal violence on social media, with a focus on Twitter. Using the Twitter API, tweets are extracted and classified according to categories of hate and intolerance, achieving a classification agreement of 54% which improves to 78% with closely related categories. This could better understand new forms of online hate that surfaced during public health discussions as related to the dynamic evolution of hatred speech (HS) on social media: it evidences the need for definitions of HS clearly stated in the form of standardization, which mostly complicate the way it can be detected and classified to improve methodologies. The benefit is that it helps to know the dynamics of HS in social media, and the drawbacks are that it poses difficulties on the definition and classification of hate speech across different contexts.

Table 2.1 Summary of existing approaches

Ref.No.	**Objective**	**Methodology**	**Result**	**Significance**	**Limitations**
[1]	An Application to Detect Cyberbullying Using Machine Learning and Deep Learning Techniques.	Using deep neural networks and word embeddings, methodology was proposed for detecting cyberbullying messages in text data. Uses CNN-BiLSTM model.	The final model has accuracy of 94.94%.	Deep learning algorithms are proven to be highly effective at text classification, with state-of-the-art outcomes on a variety of classic academic benchmark issues.	LSTM usually produces superior results, but it takes longer to process than CNN. takes longer to analyze.

[2]	Cyber Bullying Detection on Social Media using Machine learning.	Detecting cyberbullying and implement a few features with the help of a bidirectional deep learning model called BERT.	The model has achieved accuracy of 91.90%.	The result shows better accuracy when using the BERT model for sentiment analysis on the Twitter dataset.	The model shows better accuracy only for few features.
[3]	Cyberbullying Detection using Machine Learning and Deep Learning	Natural Language Processing (NLP) and machine learning models.	The model has achieved accuracy of 90% using KNN.	KNN has the benefits of being simple to use, adaptable, and requiring fewer hyperparameters.	The model faced memory and overfitting problems.
[4]	Detection of cyberbullying on social media using machine learning.	SVM and Naïve Bayes to develop predictive models for cyberbullying detection.	The model has achieved accuracy of 91% .	It works on live environment. Cyberbullying detection process is automatic and time taken for detection is less.	The model shows better accuracy only for few features.
[5]	Social Media Cyberbullying Detection using Machine Learning	The evaluation of the proposed approach on cyberbullying dataset shows that Neural Network performs better.	Model has an accuracy of 92.8%	Higher accuracy works better for larger datasets.	The model cannot perform well on smaller cyberbullying data and requires deep learning techniques to outperform machine learning approaches over larger size data.
[6]	Cyberbullying-related Hate Speech Detection Using Shallow-to-	The methodology involved in testing of three deep learning and	The model achieved accuracy of about 90%	The model shows that machine learning and deep learning can help in	The model has mainly focused on algorithm performance and didn't consider differences in

	deep Learning	six shallow learning algorithms to find patterns in cyberbullying language.		identifying cyberbullying by spotting patterns in language.	context or platform.
[7]	To explore cyberbullying and evaluate machine and deep learning models for detecting it on social media.	The methodology involves reviewing different models, focusing on challenges in creating accuracy prediction systems.	The model aim is to improve detection using advanced methods.	Emphasizes the importance of models avoiding cyberbullying and taking advantage of social media for negative purposes.	Challenges include managing massive data, affecting online behavior and context-specific aggression.
[8]	To automatically detect cyberbullying in English using advanced deep learning models and new data preprocessing methods.	The methodology applies data processing technologies and utilizes CNN and transfer learning for detecting cyberbullying in social media content.	The model aim is to improve data through advanced models	The model supports cyber centers and investigation agencies in monitoring and enhancing online safety.	The model focused only on English which may limit effectiveness for other languages or mixed language content.
[9]	To review machine learning methodologies for detecting hate speech focusing on methods and challenges.	The methodology involves supervised, unsupervised and semi supervised learning , feature engineering and tools for detecting	Describes performance improvements with advanced methods but lack specific metrics.	The model stresses the importance of detecting hate speech to reduce harm and societal division.	The model includes issues like class imbalance context sensitivity and evolving language trends.
[10]	To develop	The	The model	The model	The model

	a model for detecting cyberbullying in text data from Twitter and Wikipedia using Natural language processing (NLP) and Machine learning(ML)	methodology applied NLP and machine learning techniques to detect hate speech and personal attacks in Twitter and Wikipedia comments.	aims for up to 90% accuracy for Twitter data and 80% accuracy for Wikipedia data.	allows early detection of cyber bullying to reduce the negative impacts on victims.	challenges may involve adapting to various platforms and detecting minor or context-specific types of cyberbullying.
[11]	Deep learning approach to detect cyberbullying on twitter.	The Multi-Layer Detection (MLP) based model was used.	The proposed model achieved an accuracy of 93.2%.	The proposed model achieved a higher accuracy compared to machine learning methods used in previous studies on the same dataset	Deep learning models with more parameters could be more disadvantageous.
[12]	Cyberbullying Detection using machine learning	Combination of Natural Language Processing (NLP) and Machine Learning algorithms.	The cyberbullying detection model showed 78.5% of accuracy.	Using this model live detection of cyber bullying is predicted and alert messages are shown on the chat application.	Shows lower accuracy compared to other deep learning models.
[13]	Deep Learning Approaches for Cyberbullying Detection and Classification on Social Media.	Feature subset selection with deep learning-based CB detection and categorization (FSSDL-CBDC).	The Model showed an accuracy of 91%.	FSSDL-CBDC strategy performed significantly better in classification than the machine learning algorithms.	Unsupervised feature selection (FS) for outlier detection (OD) in large amounts of high-dimensional data that must be analyzed in near real time is challenging.

[14]	Develop a hybrid deep learning emotion classification system using multimodal data to improve emotion recognition accuracy.	The methodology involved are emotion classification, deep learning, multimodal, BERT	Achieved an F1 score of 0.90, significantly higher than the baseline model's score of 0.79.	Improves applications in customer service and mental health monitoring, contributing to advancements in AI.	The need is addressed to better emotion classification systems to include multimodal data, deal with imbalanced datasets, and employ deep learning techniques such as LSTM.
[15]	Conduct a systematic review of methods for detecting hate speech using NLP and deep learning techniques.	Utilize the PRISMA framework, including keyword selection and searches in Google Scholar and the ACM Digital Library.	Focus on evaluating the performance metrics and datasets used in existing hate speech detection approaches.	Provide a contemporary overview while emphasizing key challenges in the field of hate speech detection.	Identify a lack of comprehensive and up-to-date reviews on hate speech detection methodologies and datasets.
[16]	To provide a comprehensive overview of advancements in hate speech detection and sentiment analysis	The methodology used Hate speech detection, Sentiment analysis, Machine learning, Deep learning,Inclusive online	Deep learning models have shown superior performance compared to traditional methods.	Emphasizes the need for effective automated systems to combat hate speech and promote inclusivity online.	Identifies issues such as data imbalance, the challenge of detecting implicit hate speech, and the need for high-quality datasets in multiple languages
[17]	Intelligent multilingual cyber-hate detection in social	Systematically analyze machine learning and lexicon-based	Highlight algorithms achieving 60%-94% accuracy,	Address the rising cyber-hate issue on social media and the	Identify a lack of studies on non-English languages, particularly

	networks, focusing on taxonomy, methods, datasets, and challenges.	approaches for cyber-hate detection across languages.	with CNN and SVM showing strong performance.	need for effective multilingual detection methods.	Arabic, and the need for annotated datasets.
[18]	To enhance the detection of subtle hate speech in social media using advanced AI techniques.	Utilizes advanced natural language processing (NLP) models and multimodal analysis to interpret contextual cues in text.	Aims to improve the accuracy of hate speech detection by addressing biases and enhancing contextual understanding.	Contributes to creating safer online environments by effectively moderating harmful content while respecting free expression.	Identifies the need for more robust datasets and continuous learning mechanisms to adapt to evolving language and societal norms.
[19]	Review of relevant literature on hate speech detection techniques, maintaining quality and performance, especially in Amharic.	Collect relevant data, annotate related posts and comments, and apply machine learning algorithms such as Naive Bayes and Random Forest.	Achieve 79.83% accuracy using a word2vec embedding-based model.	Highlight the urgency of the detection of hate speech and misinformation to reduce the impact of their hindrances	Lack of benchmark datasets and generally NLP tools for Amharic, especially standardized data annotation guidelines.
[20]	Review and analyze deep learning models for effective cyberbullying detection across various data types.	Conduct a systematic review comparing different deep learning approaches and their effectiveness in detecting cyberbullying.	Models show varying accuracy, with some achieving high performance in cyberbullying detection, though	Address the growing issue of cyberbullying and its mental health impact, emphasizing the need for effective detection methods.	Identify limited application of certain deep learning techniques and the need for comprehensive datasets reflecting diverse cultural contexts.

			specific metrics are not detailed.		
[21]	Develop deep learning algorithms to efficiently detect cyberbullying from multimodal and multilingual sources across the social media base.	Use machine learning classifiers and deep learning models to recognize and classify user-generated content in multiple languages.	Hyperparameter tuning is emphasized to ensure effective classification results in deep neural networks.	Address the global issue of cyberbullying with automated detection systems across languages and modalities, enhancing online user safety.	Identify limited work on multilingual cyberbullying detection, particularly in non-English languages, highlighting the need for comprehensive solutions.
[22]	Explore recent advances in hate speech detection on Twitter by evaluating and synthesizing existing research.	Utilize Hate speech, classification, automatic detection, twitter, systematic review, natural language processing, social media.	Indicate that a perfect solution for hate speech detection remains elusive, highlighting ongoing challenges in achieving high accuracy.	Provide valuable insights and a comprehensive resource for future studies to aid in developing more effective hate speech detection models.	Identify critical challenges and opportunities for further investigation, emphasizing the need for improved methodologies and performance measures.
[23]	Platformization hate. patterns and algorithmic bias of verbal violence on social media	Utilize Twitter's API to extract tweets and classify them based on defined categories of hate and intolerance	Achieve a maximum agreement of 54% in tweet classification, increasing to 78% with closely	Highlight the emergence of new forms of online hatred during public health discussions, contributing to the understanding of HS	Identify the lack of a clear definition of Hate Speech, complicating classification and detection processes, and suggest the need for improved methodologies

			related classificati ons	dynamics in social media	

2.2 Summary: Drawback of Existing Approaches

In the annals of hand gesture-to-text conversion, a myriad of techniques has been explored, each beset with its own set of limitations. A variety of machine learning and deep learning techniques have been developed in the field of identifying hate speech and cyberbullying on social media, each with i ts own set of difficulties. The dependence on huge, excellent labeled datasets—which are frequently hard to come by—is one major problem. Furthermore, the generalizability of these models is diminished by issues with data imbalances and linguistic or cultural differences that influence the expression of harmful information. Additionally, a lot of current methods ignore multimodal data, such as images and videos, which are becoming more and more common in social media interactions, in favor of text-based content. Additionally, some models have trouble detecting harmful information across languages and situations due to their difficulties with cross-lingual detection.

Despite the improvements in detecting performance, a number of restrictions still exist. Scalability is a major issue with these models since they frequently demand large amount of computing power, which limits their applicability for broad use. Another degree of complication is added by the way hate speech and cyberbullying are changing, especially in the ever-changing social media landscape. To overcome the difficulties in identifying hazardous content across a range of mediums and languages, this emphasizes the necessity of constant progress in datasets, algorithms, and model diversity. Overcoming these obstacles by creating more resilient, multilingual, and multimodal models that can adjust to the quickly shifting terrain of harmful online content is probably the main focus of this field's future.

CHAPTER 3

PROPOSED METHOD

3.1 Problem Statement and Objectives of the Project

Problem Statement

Social media civil harassment and incitation of hatred are common problems today, especially with platforms like Twitter where inappropriate information can be spread within seconds. Such content can cause anything from mere frustrations to the division of societies through individuals in a community. And even though this issue is beginning to be understood, the current efforts to prevent and reduce cyberbullying and hate speech are predominantly manual and passive, thus failing to resolve the issue at large and in real time. This calls for sophisticated, leak-proof systems that can screen possible cases of such behavior and recommend interventions, ideally, before such behavior is exhibited.

Conventional prevention techniques against cyberbullying and hate speech mainly depend on application of keyword filtering or user's alert, which cannot deal with the complex and dynamic harmful content. Even more so, this is the case judging by the fact that there are such subtleties as coded or implicit hate speech, sarcasm, and cultural contexts, which make such conventional systems run the risk of misclassifying. Furthermore, such an approach also makes it easy for the aim of identifying harmful tweets or content to be very tedious as it will always call for a manual effort especially in incidents or conversations that lead to a lot of content generation. This indicates that there is a need for a better system, which is more affordable and capable of dealing with online interactions.

Given the rapid development of these outlets and the extensive population they cover, it becomes important to create systems capable of analyzing negative content in real time. Most of the tools usually do not provide this feature, and harmful content such as abusive tweets spreads to a great extent before they can be contained. Not responding in time. Complications when combating such forms of violence are experienced, and the public confidence in mechanisms such as Twitter and others to create safe virtual environments dwindles. In order to overcome this challenge, it is important to use modern innovations such as Natural Language Processing (NLP) and Deep Learning to improve the detection rate of these systems and decrease the time taken for the detection of such content.

The combination of two or more harmful content detection models, such as a Convolutional Neural Network (CNN) and a Long Short-Term Memory (LSTM) network, can be a game-changer in detecting and classifying harmful content. As the name suggests CNNs work well in that they are able to process images or in this case texts. The sequential feature on the other hand shows how LSTMs can easily relate to each other even if the occurrences of the series vary in time. This makes it possible to comprehends the blatant and subtle forms of Bullying and Hate speech more effectively. In conjunction with real-time data gatherers such as transforming browser interfaces into action scripts using Selenium Webdriver and other methods such as classification and sentiment analysis, these qualitative approaches and techniques build a solid infrastructure for addressing hate speech and other types of abuse on the internet.

In this project, we are building an AI system going beyond the current approaches towards Twitter abuse monitoring and mitigation. In addition to detection, the system provides built-in features that facilitate the analysis of the public sentiment and emerging trends. These features enable researchers or policymakers or organizations not just comprehend the extent of a problem but also counteract it with functional measures for fostering safer and inclusive online environments. With the help of real time insights and monitoring, this project takes a more aggressive standpoint in the fight against cyberbullying and hate speech present in the contemporary world.

Objectives of the Project

To Automate tweet collection using Selenium WebDriver for efficient extraction from specified accounts:

A unique Selenium WebDriver has been developed for this project to automate the harvesting process of tweets from certain user accounts, hashtags, or topics as efficiently as possible. The WebDriver is tasked with performing user-like activities such as scrolling and moving around the Twitter site in order to enhance data collection. This strategy makes it possible to fetch the relevant tweets as they are being posted, hence no manual inputs are required. It makes the collection of data less cumbersome, allowing the system to handle great quantities of tweets with high precision and regularity. This aspect is particularly important for later stages of analysis like inferencing, sentiment scoring, and topic segmentation.

To Utilize CNNs for text processing and LSTMs for dependency and classify tweets into relevant categories:

The project employs a combination of CNN and LSTM models to achieve high accuracy in the classification of tweets. Text images are processed by the CNNs concentrating on the diverse patterns and key areas of interest. In contrast, the LSTMs are used to model the sequence of interactions in order to comprehend the progression of ideas within the text. This makes it possible to classify most tweets under smoothly flowing categories such as neutral, hate speech or even cyberbullying. This strategy vitalizes the overall system by enhancing its ability to unmask hidden and damaging content to the utmost.

To Analyze tweet sentiment and use topic modelling to identify emotional tone and key themes:

The initiative assesses the emotions contained in the tweets to classify them into positive, negative and neutral tones. It also uses topic modelling methods to extract central ideas and currently brewing conversations out of the tweets. This two- tier approach serves to enhance understanding of the public feelings and tracking emerging issues as they arise. By looking at both the tone of voice and the subject matter of the discussions, a more rounded picture of the conversations in the cyberspace can be given. This increases the efficiency in the monitoring and counter active measures against harmful content.

To display real-time insights with a dashboard visualizing tweet categories, sentiment, and trends:

The undertaking includes a current status board that represents grouped tweets, the distribution of sentiments, and topics that are being discussed the most. This facility interface helps the users to easily navigate and study the data. The board gives a practical overview and is helpful in understanding the audience's feelings and the hot topics being discussed. It ensures fast spotting of inappropriate content and new trends.

3.2 Explanation of Architecture Diagram

3.2.1 Architecture Diagram

An architecture diagram is a representation that showcases the structure and various components of a system illustrating how they work together to achieve the systems objectives. Architecture diagrams serve as a valuable communication tool among stakeholders, providing a clear overview of the system's design and facilitating discussions about its structure and functionality.

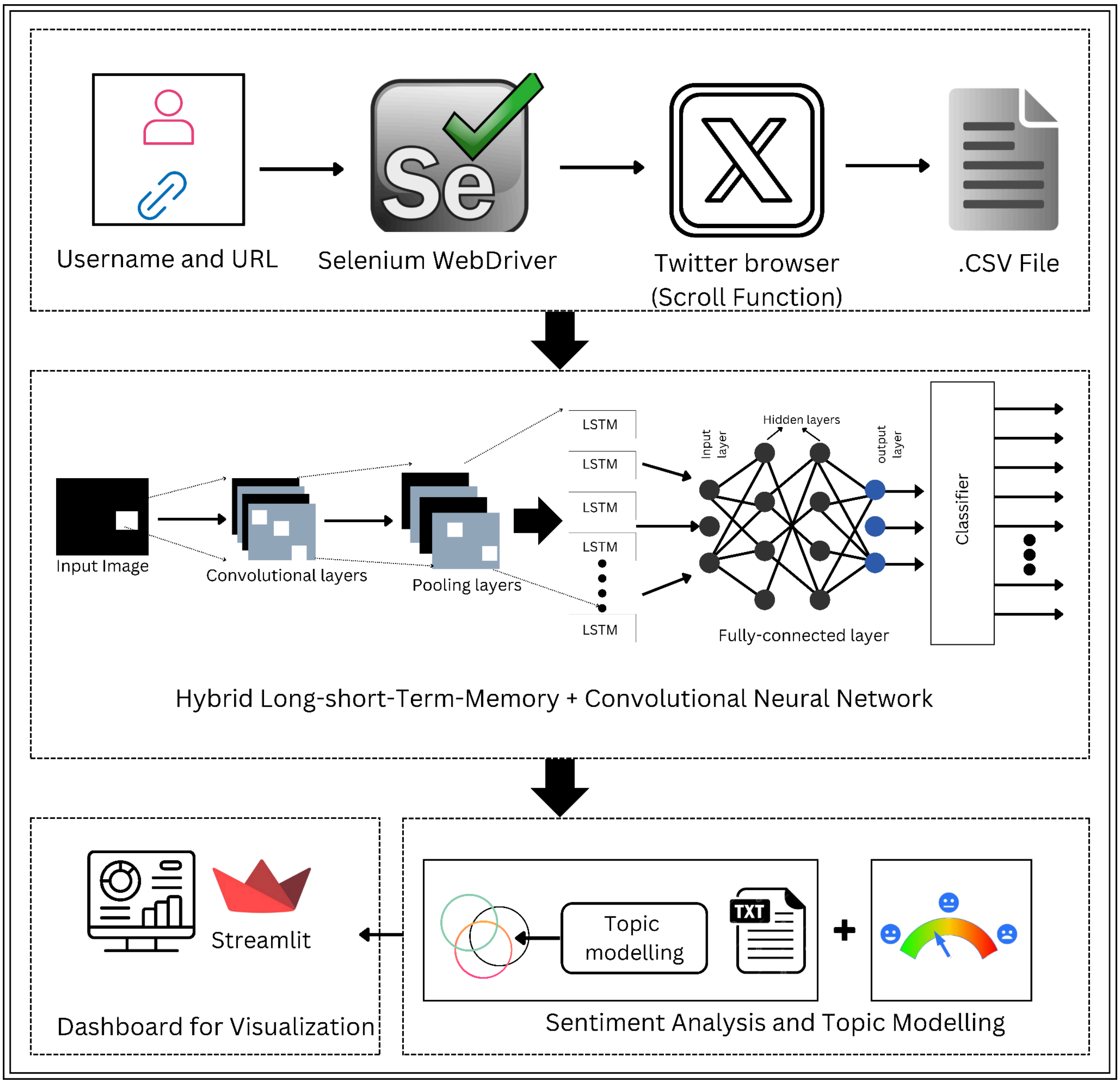


Figure 3.1 Architecture Diagram

The architecture overview shows a framework for Twitter analysis in the iContext application. The first step in the process involves introducing the username and URL, both of which are subjected to treatment through Selenium WebDriver. Selenium web scrappers, which also features a scroller, help in working with Twitter by automating the collection of numerous tweets. The gathered tweets are then placed into a structure of a CSV file for easy management of the data set in the later steps.

In the next step, a hybrid deep learning model combining Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) is used. The initial processing of the data is done by the CNN layers which concentrate on important features through the use of convolutional and pooling layers. The features are then given to the LSTM layers which are capable of processing sequences of data over time, to preserve such relationships in the data. This combined structure is capable of managing spatial and temporal variations of the tweet data making it enhances performance in tasks like topic analysis and sentiment classification.

Ultimately, the after treatment data on the statistics is presented and explored by the users in a form of a dashboard using Streamlit. The results of the sentiment analysis are presented alongside the results of the topic modelling, showing the overall emotional tone, and the main themes found in the tweets, respectively. Topic modelling is the processes of clustering and categorizations of tweets with respect to the topical content of the tweets, while sentiment analysis measures the mood of the content and divides it into a range of, positive, negative or neutral scores. This architecture is designed to collect Twitter data, perform advanced modelling and visualize the results easily making it useful for social media analysis.

3.2.2 Modules-Connectivity Diagram

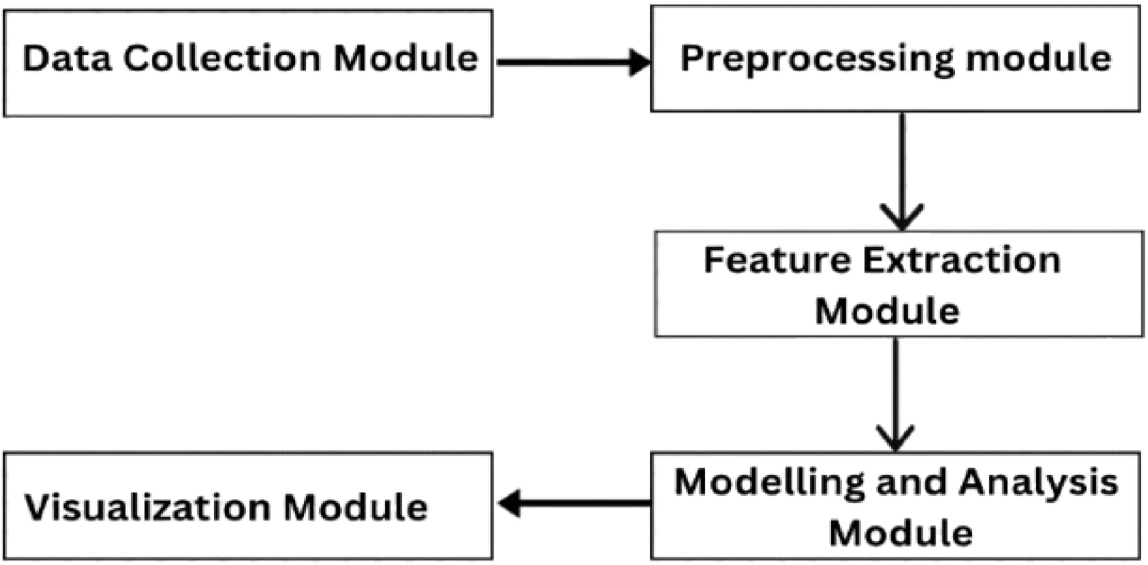


Figure 3.2 Modules-Connectivity Diagram

The Data Collection Module serves as the building block for the Twitter data analysis project. To facilitate this process, the module incorporates the use of the twitter API through the Selenium WebDriver. The WebDriver is provided with a Twitter username and a URL in order to navigate to the profile and then performs a scroll action to display the complete timeline of the user. This scrolling function is very important in this regard as it helps to collect every tweet instead of only a few ones that are shown at the screen. It subsequently goes on to calculate the text, time, and other parameters of tweets between given time intervals and saves the information into a CSV file. As such this automating process of collecting the data helps in achieving the collection of big amounts of data making it easy and suitable for the next stages of the project.

After the gathering of information, the Preprocessing & Feature Extraction Module readies the tweet data for analysis by reorganizing the information. Clean-up stage comes first where text is prepared without elements such as URLS, special characters, and even smiling faces which may interfere with the model in learning patterns. Afterwards, case normalization comes into play with all the text being changed to lower case. Furthermore, tokenization divides the tweets into their constituent parts, which are referred to as words or tokens, hence facilitating the processing of the data. Stop-word removal, stemming or lemmatization later come into place to process the dataset by cleaning it from extraneous speech. At this point in the process, the convolutional layers of the Convolutional Neural Networks (CNN) are integrated with this framework in the feature extraction phase, which processes input text in order to find patterns of certain words and local characteristics within the text. And then pooling layers for dimensionality reduction and feature emphasis so that a dataset that is rich in features is generated in time for the next module's analysis.

The Modeling and Analysis Module uses machine learning methodologies for doing sentiment analysis and topic modelling on characteristics gathered from the tweets. This module utilizes both the advantages of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. The CNNs layers of the text model concentrate on the spatial constituent of the text to capture certain local arrangements of words that depict either messages' sentiments or topical relevance. The LSTM model structure provides a way to store the sequence of input data, allows different time patterns and relationships between the words to be learned. After the sequences have gone through the LSTM layers, a hundred percent connected layer spans the features and the next layer predicts the output. It determines, the sentiments of every tweet as positive, neutral or, negative and for every topic there is a subsection that is devoted to a specific topic modeling. The combination of CNN and LSTM contributes to the success of this mixed model in high level pondering structure and content of the tweets.

At last, the Visualization Module makes use of Streamlit to exhibit the analysis findings in a formatted manner for the users. This module takes in the output from the Modeling &

Analysis Module, specifically the sentiment scores and topic clusters, and presents it in an interactive dashboard format. In this case, users get to analyze the various trends in sentiments, how the various tweets are thematically clustered and other data driven analyses. Illustrating the collected data into a Streamlit dashboard makes it even easier since it provides a clear visualization of how the data could be presented with concentrated calculations on the key points and it is possible to navigate the data easily. This final component is very advantageous because it visualizes complex analysis results, thereby helping the users understand how the information or data collected relates to the various identified trends, patterns and the overall mood from the tweets collected.

3.2.3 Software and Hardware Requirements

Software Requirements

1. Programming Language:
   Python 3.8+
2. Libraries/Frameworks:
   - ○ Web Scraping:
     Selenium
     BeautifulSoup (optional)
   - ○ Data Processing:
     pandas
     numpy
   - ○ Natural Language Processing (NLP):
     nltk
     gensim
     textblob
     vaderSentiment
     Hugging Face Transformers
   - ○ Machine Learning/Deep Learning:
     TensorFlow or PyTorch
     scikit-learn
   - ○ Visualisation:
     matplotlib

seaborn

plotly

wordcloud

- ○ Dashboard Development:

  Streamlit

3. Browser Automation:

   Google Chrome or Firefox (for Selenium)

   Corresponding WebDriver (e.g chromedriver for Chrome)

4. Environment Management:

   Virtualenv for creating isolated environments.

5. Text Editors/IDEs:

   VS Code

6. Operating System:

   Compatible with Windows 10/11, macOS, or Linux.

7. Cloud Platforms:

   Streamlit Community Cloud for free dashboard hosting.

Hardware Requirements

1. Processor:

   Minimum: Intel Core i5 or AMD equivalent.

   Recommended: Intel Core i7/i9 or AMD Ryzen 7/9.

2. RAM:

   Minimum: 8 GB

   Recommended: 16 GB or higher (for handling large datasets and training models efficiently).

3. Storage:

   Minimum: 256 GB SSD

   Recommended: 512 GB SSD or more for faster file read/write and data storage.

4. Graphics Card (for Deep Learning):

   Integrated Graphics (for basic model training and operations).

5. Display:

1080p or higher resolution for clear data visualisation.

6. Internet Connection:

   Stable internet with sufficient bandwidth for downloading libraries,
   data scraping, and API usage.

3.3 Modules and their Description

1. Data Collection Module

There is a component in the system called the Data Collection Module which is designed to obtain tweets from Twitter with the help of the Selenium Web Driver. First, it takes basic inputs like the Twitter user & the URL to load the page and proceed with the automation of fetching the needed information. The module has a function to scroll down to the end of the page to ensure that all the tweets and not only the relevant ones are fetched. The data obtained through this process contains information about the tweet, the time at which the tweet was made and other details which are then organized in a readable manner in a file such as a CSV for easy analysis. This method of collecting data is automated which makes it less laborsome and easier to obtain large quantities of information

2. Preprocessing Module

The Preprocessing Module obtains raw data from the Data Collection Module and processes it for analysis by transforming the unprocessed text into a properly organized one. This module first eliminates extraneous components such as URLs, hashtags, user mentions, punctuation, and icons that could otherwise help create ‘noise’ in the model. Thereafter, all the text is transformed into lower case characters in a bid to normalize the data and enhance uniformity. Tokinisation splits a tweet into smaller means or tockens which in this case are individual words making it easy to handle such data especially when advancing to further processing stages. Additional processes, such as removing stop words and stemming or lemmatization, are used to condense each tweet to its crucial parts, thereby helping the model constrain attention to only necessary words. With this dataset structured and cleaned, accurate feature extraction is guaranteed.

3. Feature Extraction Module

The Feature Extraction Module is responsible for converting the preprocessed text into feature representations that can be utilized by machine learning models. In this project, Convolutional Neural Networks (CNNs) are used to pull out spatial structures from text data. The given CNN layers are designed to learn local parts of the data, e.g. some common sequences of words or some patterns which are useful for identifying the sentiment and the theme. Then come pooling layers which reduce these features so as to make them less descriptive while retaining most of the crucial content so as to ease the process of analysis that comes afterwards. Here over the dimensionality-reduced feature set, the Modeling & Analysis Module comes into play, where this feature set is utilized when enhancing the performance of the LSTM layers and the classifier.

4. Modelling and Analysis Module

The Modeling & Analysis Module employs a combination of Long Short Term Memory (LSTM) and Convolutional Neural Networks (CNN) to analyze the sentiment and topic in the dataset of tweets. The LSTM layers maintain the sequential order of the tweets, that is, it understands the patterns over time, while the fused layer combines all the features to carry out the predictions. Later the classifier predicts the sentiment for the tweet as positive, neutral or negative and finds a corresponding topic for it. Such a model enhances the systems capability to recognize and decipher more complex lingual patterns for better performance in sentiment analysis and topic modelling.

5. Visulalization Module

The results of the analysis are presented via an interactive Streamlit dashboard in the Visualization Module, allowing for easy interpretation of analysis insights by the users. This module presents the sentiment scores, enabling the users to measure the overall sentiment of the tweets, as well as topic clusters portraying how common certain topics are. Users go into the dashboard and can also choose sentiments or topics to learn the trends and patterns in detail. This visualization tool is the last component of the project and it aims to provide a simple way to navigate through the analyzed data.

Evaluation of the results: Following the model's classification, its accuracy undergoes evaluation using the sklearn module. If the accuracy falls below satisfactory levels, adjustments are initiated to enhance model performance. Potential modifications include augmenting the dataset size or increasing the number of decision trees within the classifier. By expanding the dataset, the model gains exposure to a wider range of examples, facilitating more robust learning. Similarly, augmenting decision trees enhances the model's complexity, potentially improving its ability to capture intricate patterns within the data. These iterative adjustments aim to optimize model accuracy and ensure reliable performance in classifying input data.

3.4 Requirements Engineering

Functional Requirements

a. Data gathering:Using either the Selenium program or Twitter's software development kit, gather tweets, and then save them in a Comma Separated Values (CSV) for further analysis.

b. Data Preparation:Edit the text of tweets and get it ready for analysis by cleansing, and putting the text into proper shape through the use of topic modelling.

c. Sentiment Evaluation:For multiple analysis techniques in classifying tweets, enhance accuracy and classify them according to their sentiments.

d. Thematic Analysis:Due to the large volume of tweets, employ a Latent Dirichlet allocation (LDA) model to categorize the tweets based on the major topics represented in the tweets.

e. Visualization Dashboard:Deliver a user-friendly and interactive Streamlit dashboard that can be used to display and filter insights on sentiments and topics.

f. Storage and Export:Organize the cleaned data for easy storage in CSV files and allow for storage in a manner that is quick for the user to use.

g. Model Training:Create a CNN + LSTM model and fit the labelled datasets for training of the multi-class tweets classification.

h. Forecasting:Employ the fitted model in the appraisal of emotions and subjects of fresh tweets.

Non-Functional Requirements

a. Performance:The system must process at least 100 tweets per minute during scraping and preprocessing, with predictions made in under 1 second per tweet.

b. Scalability:The system should handle up to 100,000 tweets without performance degradation and support future model expansions.

c. Usability:The dashboard must be intuitive and easy to use, even for users with limited technical knowledge.

d. Reliability:The system must handle failures gracefully and ensure data integrity during storage and export operations.

e. Compatibility:The system must be compatible with Windows, macOS, and Linux, and support major browsers for Selenium.

h.Maintainability:The codebase must be modular and well-documented to facilitate debugging and future updates.

i.Security:Sensitive data like Twitter credentials must be securely handled, and the system must comply with Twitter's API policies.

j.Efficiency:The system should leverage GPU acceleration for deep learning and optimize resource usage to avoid system lag.

k.Accessibility:The dashboard must be accessible via any modern browser and responsive across different screen resolutions.

3.5 Analysis and Design through UML

3.5.1 Class Diagram

A class diagram is a fundamental component of Unified Modeling Language (UML) used in software engineering to illustrate the static structure and relationships within a system. It provides a visual representation of the classes in a system, their attributes, methods, and the associations between them. Here's a breakdown of the key elements found in a class diagram

Class

The central building block of a class diagram, representing a template or blueprint for creating objects. It encapsulates both data (attributes) and behavior (methods) related to a specific concept or entity in the system.

Attributes

Characteristics or properties of a class that define its state. Attributes are typically depicted as variables within a class and describe the data that objects instantiated from that class will hold.

Methods, Functions or operations associated with a class that define its behavior. Methods specify what actions an object of the class can perform and how it interacts with other objects. Figure 3.3 Shows the Class Diagram of the TweetScraper designed to retrieve tweets for a given Twitter account through a profile_url, the TweetScraper class is specialized in scraping Twitter profiles. It has a scrape Tweets() method that retrieves a large number of tweets for all-round processing. This class is the focal point of the system ensuring the precision and effectiveness of the process.

Tweet: The class Tweet acts as a buffer to share data before and after the tweets are processed. It keeps the attributes which get populated like tweet_text for the actual tweet content cleaned_text for the ready but not yet analyzed data and sentiment or topic for the analysis derived from the data. The organization of data across classes ensures that data for specific entities like tweets is kept properly for use in future tasks.

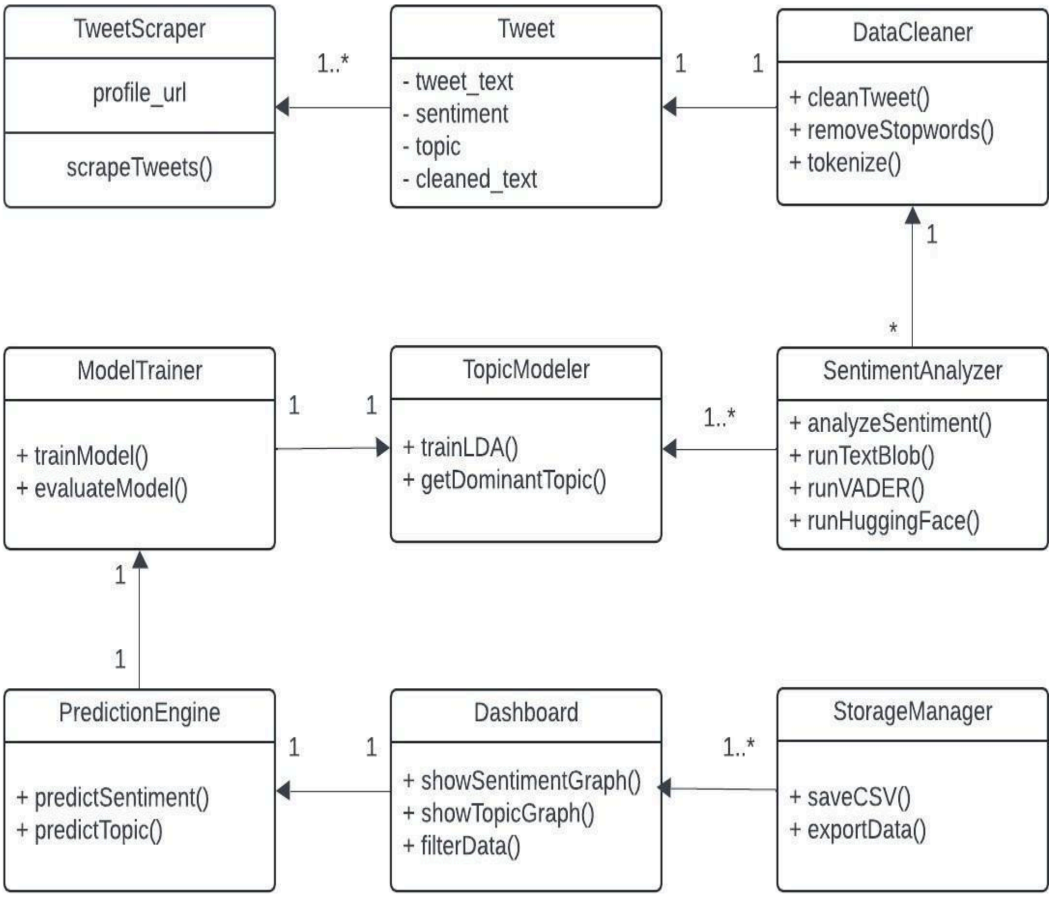


Figure 3.3 Class Diagram

DataCleaner: This class takes care of all preliminary preparations that are required on the text content of the tweets before the actual analysis is done. It contains functions such as cleanTweet() which takes care of cleaning the tweet by eliminating unnecessary text, removeStopwords() which removes words that are inconsequential, and tokenize() which helps to break down the text into its constituents. This ensures that the tweets are clean and ready for sentiment or topic modelling.

ModelTrainer: The ModelTrainer class encompasses all functions that pertain to the training and testing of machine learning basing technologies deployed in the system. Through methods such as trainModel() and evaluateModel(), classifiers like the one to classify tweets is tweaked until optimal performance is obtained. This ensures that the system developed has adequate capabilities in making predictions especially on sentiments and topics defined. SentimentAnalyzer:The SentimentAnalyzer class refers to classes that provide means and ways to do sentiment analysis on tweets. It employs numerous methods like runTextBlob(), runVADER(), runHuggingFace(), to perform analysis of sentiment matters. This class is an important addition to the sentiment detection capabilities of the system in that it enhances the processes while making them more efficient and precise.

TopicModeler: The TopicModeler class participates in figuring out which topics prevail in a set of tweets. It deploys functions such as trainLDA() for the purpose of extracting topics, and getDominantTopic() for the task of assigning specific topics to given tweets. This allows thematic understanding and insights to be drawn from the tweets data.

PredictionEngine:The PredictionEngine class combines the predictions from sentiment and topic analysis. Using methods such as predictSentiment() and predictTopic(), it employs models built on previously analyzed tweets, to categorize fresh tweets. This class guarantees predictive capabilities in the system at all times.

Dashboard: This process includes the practical graphical representation of the already processed findings under the Dashboard class. It has methods such as showSentimentGraph() and showTopicGraph() for good visualization of statistics. This class is user-friendly clear and an enhanced visual representation in a good interactive way will contribute towards even better understanding.

StorageManager: The StorageManager class is responsible for performing tasks related to data persistence, that includes saving and exporting results. Its methods, saveCSV() and exportData(), permit saving of processed data for later use. This class enhances the management and access of data.

3.5.2 Sequence Diagram

The different Lifelines in the diagram are

1. User.
2. Classifier.
3. Programmer.

The sequence diagram in Figure 3.4 shown depicts the working of the system from a user's perspective which involves the analysis and visualization of tweets. The process starts when an actor (user) inputs the URL or user name in order to begin fetching the data. This extraction is done using Selenium which fetches the tweets from the provided source and saves them in a specific layout, for instance, a CSV file. In this respect, the diagram illustrates how the user, data extraction module and the other stages of the analysis pipe communicate to each other in the description of the system in a sequential manner.

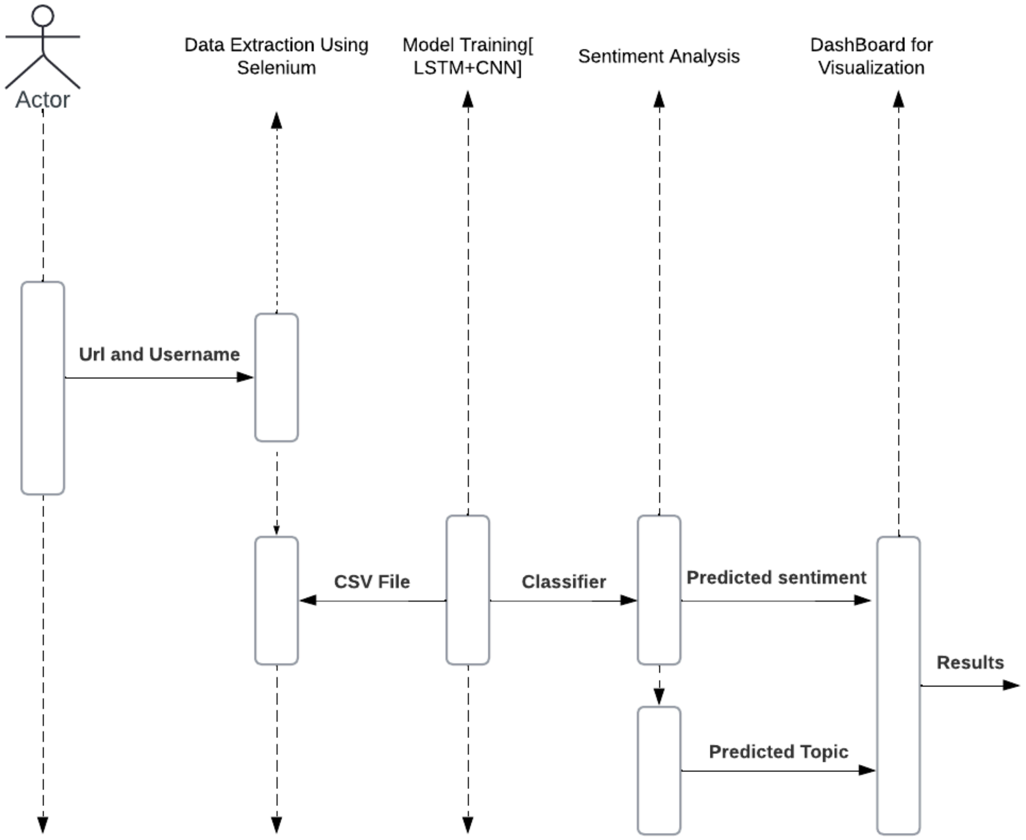


Figure 3.4 Sequence Diagram

When the twits are collected, those against the columns tessellation are used as inputs for the structural model building and classification LSTM and CNN. This is data that the classifier classifies into sentiments and topics for every tweet and these two aspects are the main products of this entire process. The sentiments and topics are then predicted and sent for visualization on a dashboard so that the user can see and understand the results. This one-way pathway of interactions is helpful in clarifying how the basic components and the intermediate components of the system are transformed into useful components and information.

3.5.3 Use Case Diagram

A user case diagram provides an easy representation of the system and its users, which visually displays interactions between different elements. It provides an overview of the events that occur in the system and its flow but does not give detailed information on how they are implemented.

The Actors

1. User.
2. Programmer.

The Use Cases

1. Sentiment Analysis.
2. Topic Modelling.
3. Data Preprocessing.
4. Predicting Topic.
5. Predicting Sentiment.
6. Visualization of results

The use case diagram depicts the relationships of users with a particular system that is being developed for the purpose of efficient data processing and visualization, in this case, a tweet analysis system. As the primary actor, a user is the one who works with the system to accomplish several tasks, including, but not limited to, collecting and saving tweets for further analysis, preparing the content for processing, teaching classification algorithms and performing analysis on sociopolitical issues in the digital environment, such as sentiment or thematic analysis. All of these use cases are well-defined and correspond to specific functional capabilities of the system showing that the analytics operations supported by the system are more than one and can be multiple.

Figure 3.5 Explains the connection between the use cases and the relationships between use cases and the actors.Conversely, the system offers outputs like sentiment analysis, topic classification, and result representation so that the users can derive useful information from the system. The secondary actor i.e. the system administrator or a stakeholder could make use of the final outputs meaning classified topics and sentiment analysis for making some decisions. The representation that focuses on interactions illustrates the importance of the

Figure 3.5 Use Case

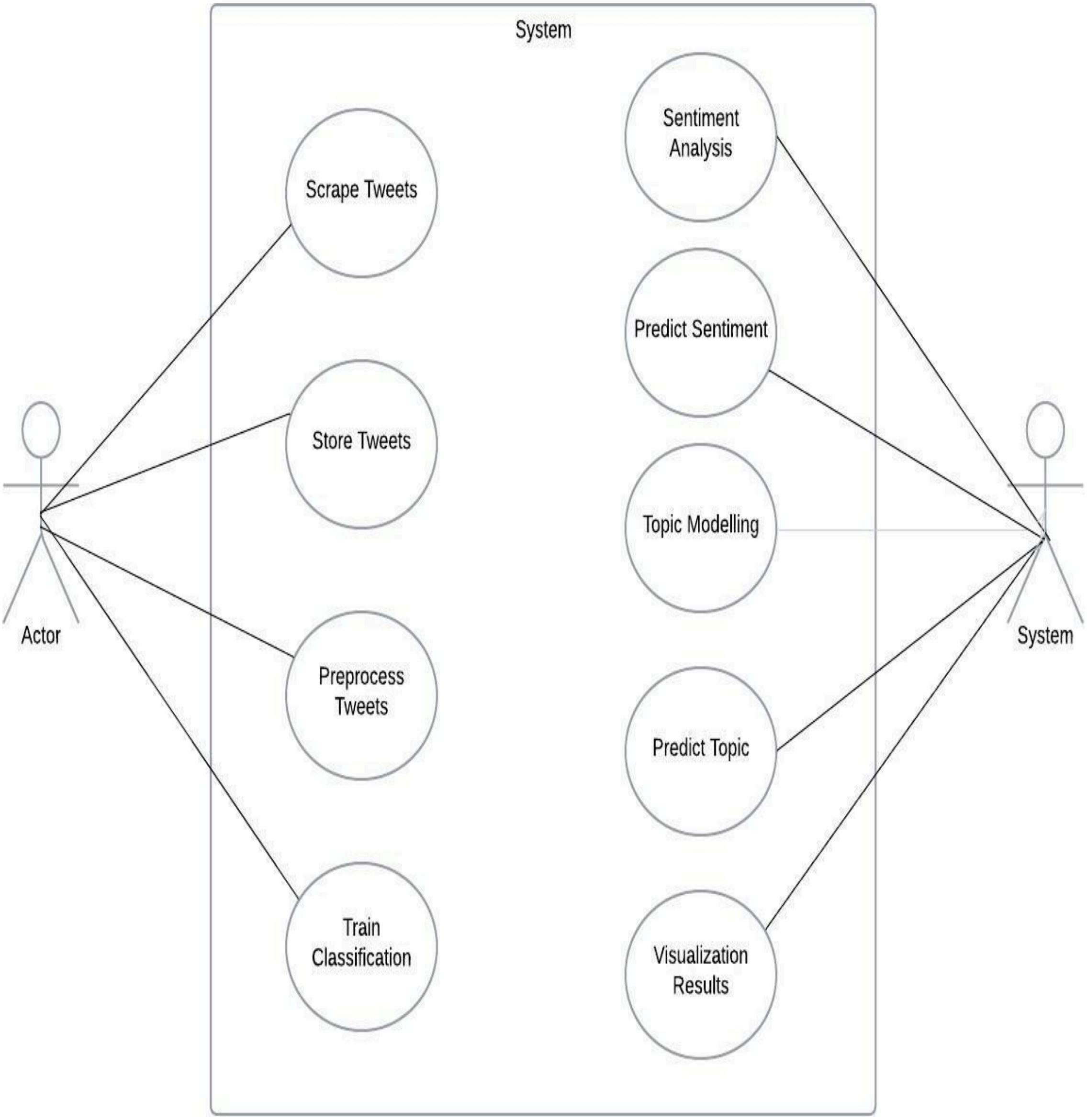


Figure 3.5 Use Case Diagram

Figure 3.5 Explains the connection between the use cases and the relationships between use cases and the actors.Conversely, the system offers outputs like sentiment analysis, topic classification, and result representation so that the users can derive useful information from the system. The secondary actor i.e. the system administrator or a stakeholder could make use of the final outputs meaning classified topics and sentiment analysis for making some decisions. The representation that focuses on interactions illustrates the importance of the system for analysis and insights generation showing how social media content is analyzed and why it is important.

3.5.4 Activity Diagram

The flowchart described a robust process to explore tweets by scraping, preprocessing, training the model, classifying sentiment, and topic modeling. It initiates with the option of scraping tweets off the platform, these in structured form, and preprocessing the content for further analysis. Preprocessing often involves cleaning up the text and preparing it to be fed into a machine learning or NLP model as an input. The chart shows lines for different operations, but with such a chart, one is able to do some of the tasks, for example, training on labelled data or sentiment analysis.

Figure 3.6 Explains the execution flow of the program.Another important feature of the flowchart is to extract insights through topic modeling as well as sentiment classification. After preprocessing, the system performs such operations as removing stopwords, tokenizing words, and analyzing sentiment in the tweets it processes. The system uses topic modeling techniques to assign dominant topics to the tweet and stores the output with both sentiments and topics in a CSV file. The insights are finally visualized on a dashboard, making it easy for the users to interpret results intuitively. This is an end-to-end workflow that efficiently enables the analysis of tweets to detect sentiment and understand what's trending through the key themes.

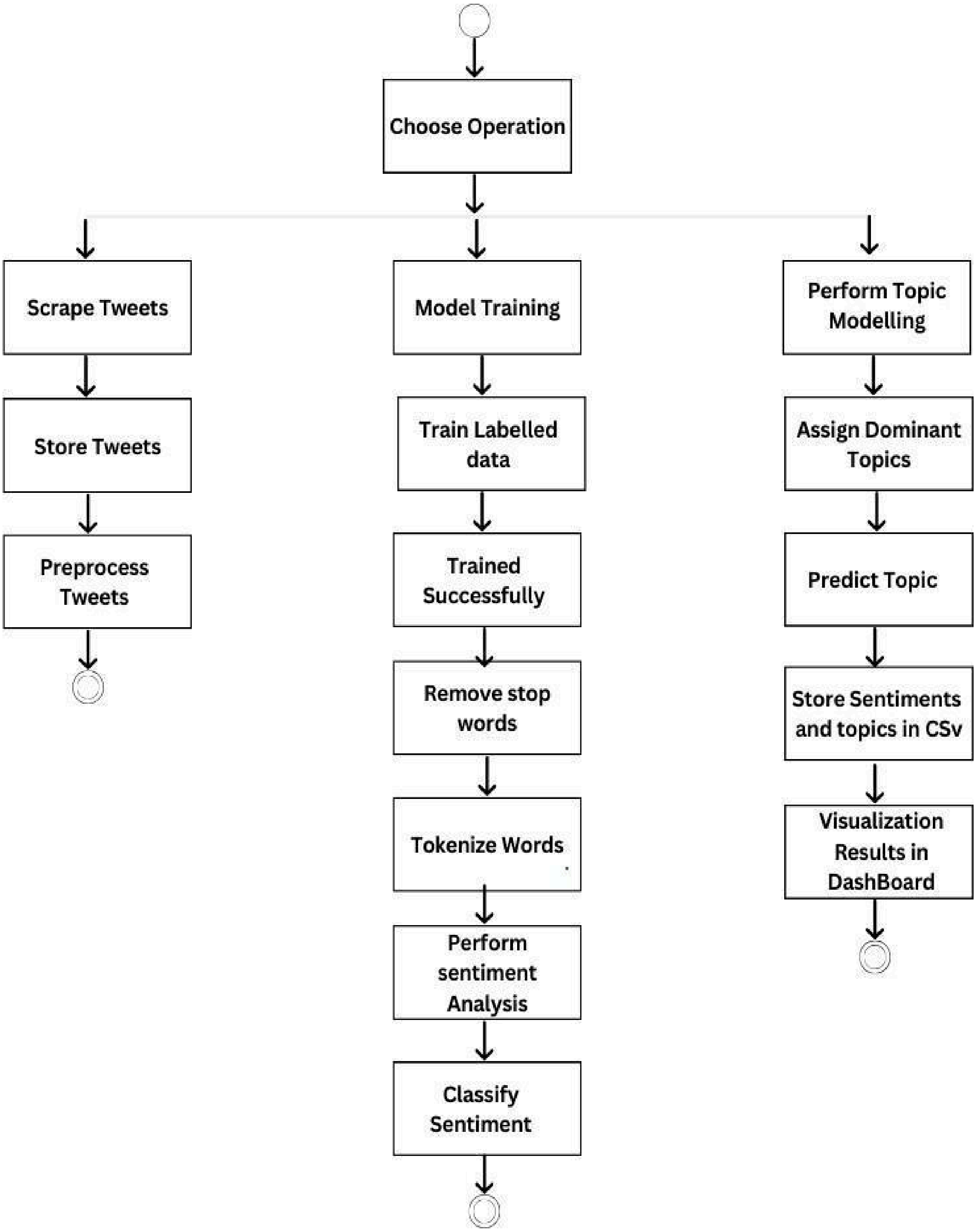


Figure 3.6 Activity Diagram

CHAPTER 4

RESULTS & DISCUSSIONS

4.1 Description of Dataset

In our project, the dataset used consists of a collection of tweets scraped from specified Twitter profiles. The data was gathered using the Selenium WebDriver, capturing tweets from various users based on predefined keywords or hashtags. This dataset includes a range of tweet content, such as text, timestamps, user information (e.g username, user location) and metadata related to the tweet (e.g retweets, likes, and tweet ID). The dataset is primarily composed of textual data, which forms the basis for sentiment analysis and topic modelling. For sentiment analysis, the dataset contains tweets with varied sentiments, such as positive, negative, and neutral tones, which were manually labelled or inferred using sentiment analysis tools. The dataset includes tweets about a wide range of topics, ensuring diverse sentiment expressions that allow the models to train effectively across different emotional tones.

For topic modelling, the dataset is rich in textual diversity, including tweets related to current events, personal opinions, products, services, and general discussions. This broad range of topics allows the LDA model to identify and group tweets into different clusters based on shared themes or subject matter. This diversity in content makes the dataset well-suited for evaluating the effectiveness of both sentiment analysis and topic modelling techniques.

The dataset is also preprocessed to remove any irrelevant content, such as URLs, mentions, hashtags, and non-alphanumeric characters, ensuring that the data used for analysis is clean and focused on the core message of each tweet. The final dataset used in the project was stored in CSV format, containing columns for tweet text, sentiment labels, and topic categories, making it ready for use in training the models and performing predictions.

4.2 Detailed Description about the Experimental Results

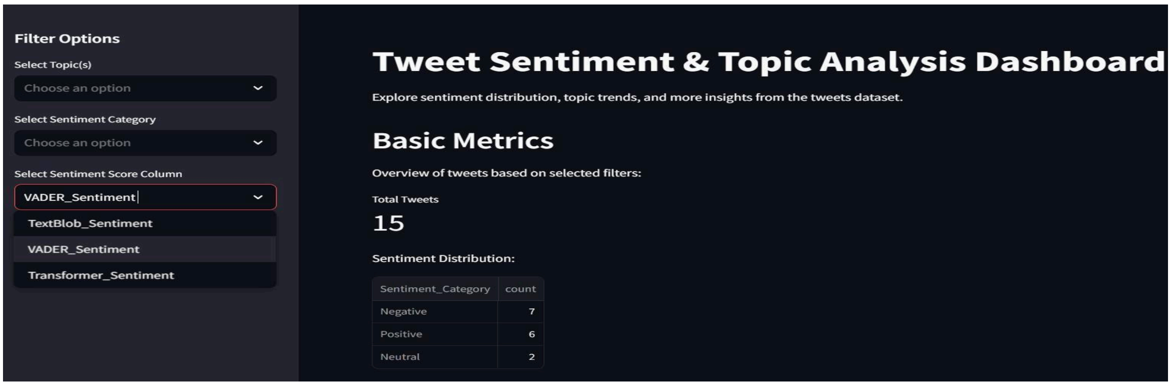
The experimental results of our project focused on analyzing and processing the dataset of tweets, applying various machine learning models for sentiment analysis and topic modelling, and evaluating the performance of these models in real-world tweet data. The models implemented in the project were TextBlob, VADER, and Hugging Face for sentiment classification, as well as an LDA (Latent Dirichlet Allocation) model for topic extraction.

For sentiment analysis, the TextBlob, VADER, and Hugging Face models produced promising results when compared with each other. VADER, being optimized for social media text, performed particularly well with tweets containing emoticons, slang, or informal language. TextBlob also provided reasonable accuracy, though it showed slightly less performance in tweets with non-standard spelling or grammar. Hugging Face's transformer-based model outperformed both, showing the highest accuracy, especially for tweets with more complex linguistic structures. The sentiment analysis was evaluated using accuracy, precision, recall, and F1-score, all of which showed high values, especially for positive and negative sentiments. Neutral sentiment tweets were slightly harder to classify, as expected.

In the topic modelling phase, the LDA model was used to identify the dominant topics in the tweet corpus. The topics were generally coherent, categorizing tweets into recognizable clusters, such as technology, politics, and entertainment, with some overlap between topics. The LDA model's effectiveness was assessed by looking at topic coherence scores and through manual inspection of the topics extracted from the tweet text. Although the topics were generally interpretable, some more ambiguous tweets resulted in topics that required further tuning of hyperparameters to improve accuracy.

The performance of the entire system, including sentiment analysis, topic modeling, and the visualization dashboard, was tested using various real-time datasets. The dashboard, built using Streamlit, offered a user-friendly interface to filter and visualize sentiments and topics over time. Visualization of sentiment distribution and topic trends helped users gain insights into the evolving nature of tweets about certain issues. The system’s efficiency in processing at least 100 tweets per minute was confirmed during testing, and the overall accuracy of the sentiment and topic classification models showed that the system could deliver meaningful results on large tweet datasets with low latency, making it suitable for real-time applications.

Experimental Results on Test Data



Filter Options

Select Topic(s)

Choose an option

Select Sentiment Category

Choose an option

Select Sentiment Score Column

VADER_Sentiment

- TextBlob_Sentiment
- VADER_Sentiment
- Transformer_Sentiment

Tweet Sentiment & Topic Analysis Dashboard

Explore sentiment distribution, topic trends, and more insights from the tweets dataset.

Basic Metrics

Overview of tweets based on selected filters:

Total Tweets

15

Sentiment Distribution:

Sentiment_Category	count
Negative	7
Positive	6
Neutral	2

Figure 4.1 Tweet Sentiment and Topic Analysis Dashboard

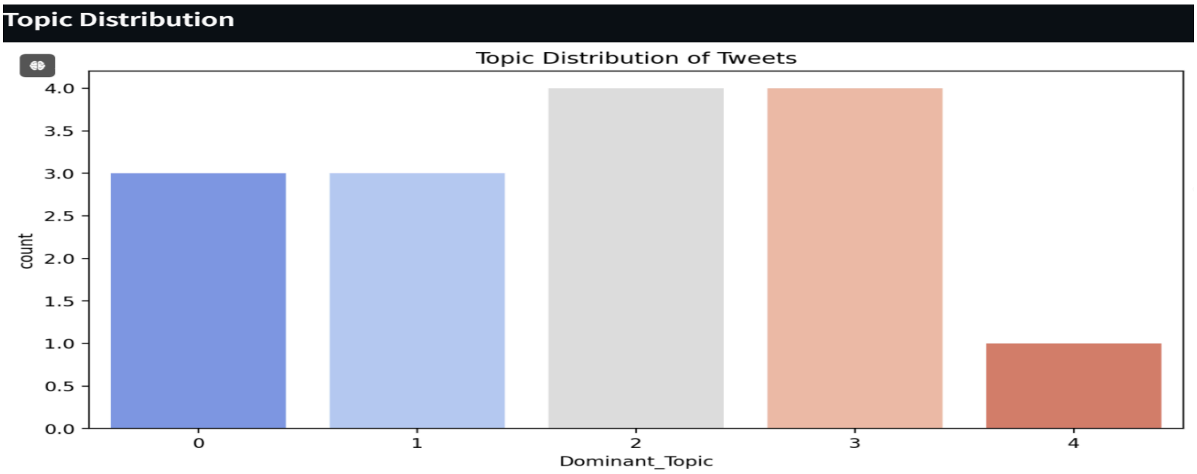


Figure 4.2 Topic Distribution of Tweets

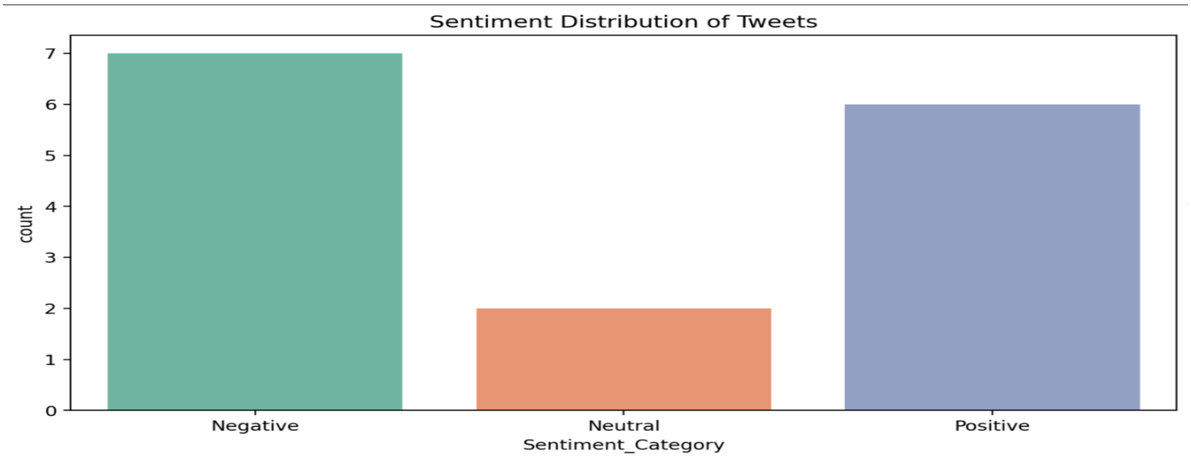


Figure 4.3 Sentiment Distribution of Tweets

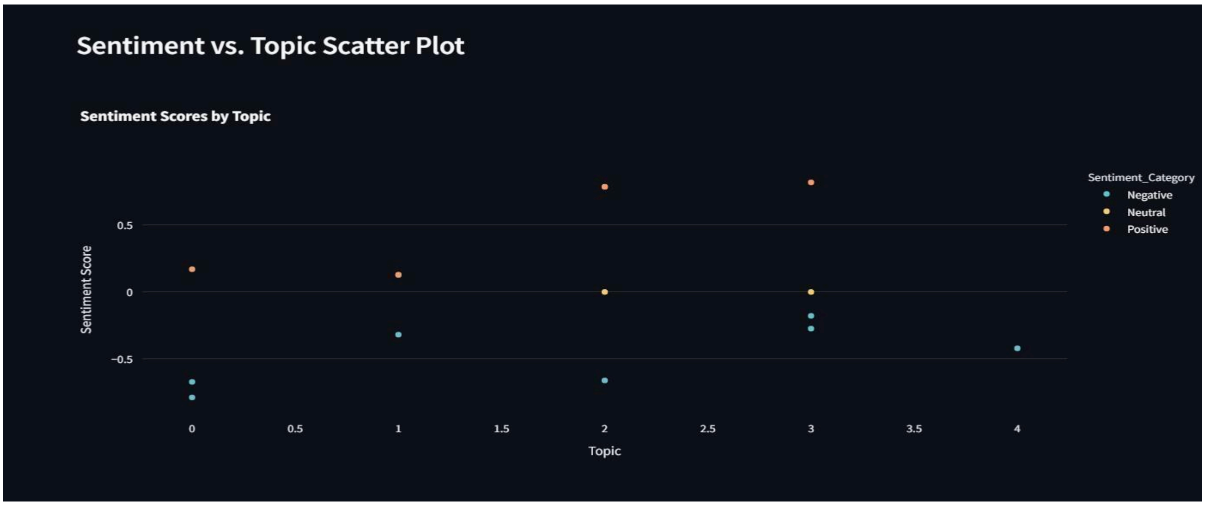
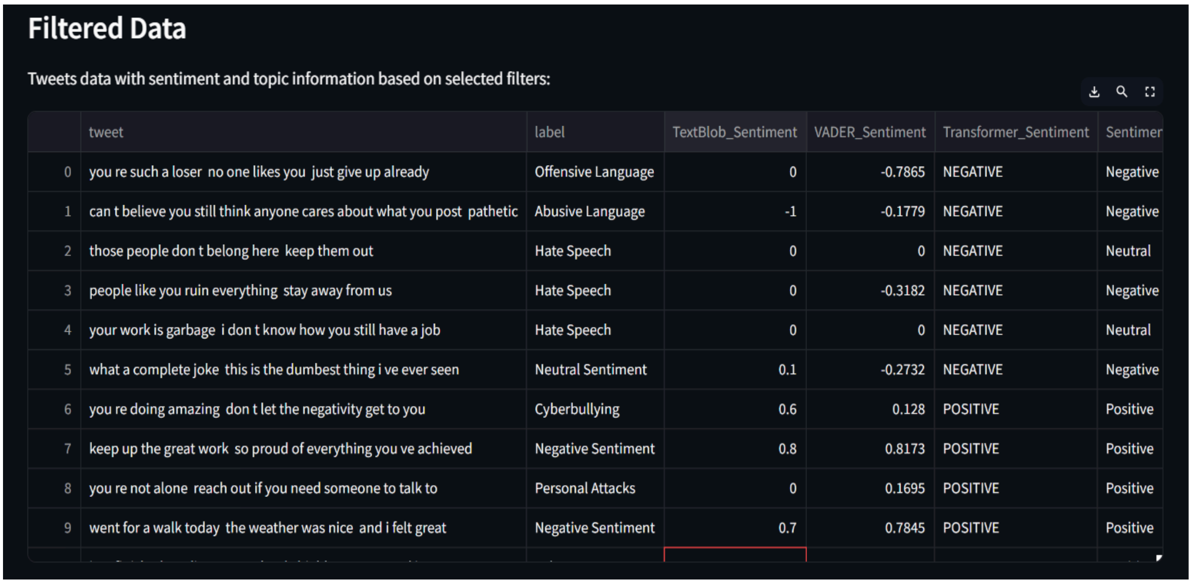


Figure 4.4 Sentiment vs Topic Scatter Plot



Filtered Data

Tweets data with sentiment and topic information based on selected filters:

	tweet	label	TextBlob_Sentiment	VADER_Sentiment	Transformer_Sentiment	Sentimer
0	you re such a loser no one likes you just give up already	Offensive Language	0	-0.7865	NEGATIVE	Negative
1	can t believe you still think anyone cares about what you post pathetic	Abusive Language	-1	-0.1779	NEGATIVE	Negative
2	those people don t belong here keep them out	Hate Speech	0	0	NEGATIVE	Neutral
3	people like you ruin everything stay away from us	Hate Speech	0	-0.3182	NEGATIVE	Negative
4	your work is garbage i don t know how you still have a job	Hate Speech	0	0	NEGATIVE	Neutral
5	what a complete joke this is the dumbest thing i ve ever seen	Neutral Sentiment	0.1	-0.2732	NEGATIVE	Negative
6	you re doing amazing don t let the negativity get to you	Cyberbullying	0.6	0.128	POSITIVE	Positive
7	keep up the great work so proud of everything you ve achieved	Negative Sentiment	0.8	0.8173	POSITIVE	Positive
8	you re not alone reach out if you need someone to talk to	Personal Attacks	0	0.1695	POSITIVE	Positive
9	went for a walk today the weather was nice and i felt great	Negative Sentiment	0.7	0.7845	POSITIVE	Positive

Figure 4.5 Filtered Data

Evaluation metrics

Our project uses numerous measures to measure the quality of the cyberbullying and hate speech detection system. First of all, the parameters of accuracy and loss during both training and validation, are used to evaluate success of the CNN-LSTM model. Accuracy refers to the extent to which the model is able to correctly assign labels, whereas loss is the measure of how far off the predicted labels are. These metrics can help in going blind in the comprehension of the performance and time taken by the model in learning, more so where there are several classes to predict. Moreover, confusion matrices help to reveal the primary, secondary and tertiary detection rates, and false detection rates measured for all the ten classes revealing the patterns of apartheid, and overall performance by class.

For the purpose of sentiment analysis, the team uses three different methods: TextBlob, VADER, and a Transformer-based sentiment model. Each approach offers something different where TextBlob gives scores based on polarity, VADER gives scores based on compound sentiment whereas Transformers give inexact scores in terms of sentiments. These are compared on the other hand in measuring the strength and validity of the sentiment classification. In addition, the sentiment types (positive, negative, neutral) which are assigned based on VADER's scores are vice-validated with the output from the other tools to enhance the accuracy of final sentiments classifications.

In topic modeling, it's common practice to calculate coherence scores for the topics generated via Latent Dirichlet Allocation (LDA), as a means of assessing their quality. Coherence measures how comprehensible and semantically related all the words - that form individual topics - are to each other. Coupled with tokenization and preprocessing steps, this ensures that the topics extracted from the tweets are useful and relevant. This is further complemented by the use of bar charts and word clouds in the Streamlit dashboard which provides visual insights of the data making the metrics more practical to the users. **Explanation of output graphs and filtered data**

1.Distribution of Categorized Tweets (Bar Chart)

Objective: This shows how many tweets have been placed in each category of tweets, here neutral, hate speech and cyber bullying.

Insights: This helps in understanding how much content in the given set is harmful, showing which category takes the most portion in the sample content.

2.Sentiment Analysis (Pie / Bar Chart)

Objective: Analysis of sentiment is conducted to get the percentage of the positive, neutral and negative sentiment in the overall analyzed tweets.

Insights: This provides the overall sentiment trend which shows how the discussions in the dataset are up to that point.

3.Topics in Focus (Word Cloud)

Objective: This is performed to show the keywords or phrases occurring the most within the tweets of the specific category or sentiment.

Insights: This showcases important concepts and areas but also facilitates the identification of topics like cyberbullying or hate speech which are easily found in the dataset.

Sentiment versus Topic Scatter Plot – Definition and uses: This plot represents the sentiment scores (in terms positive, negative or neutral) across major topics in the dataset. X-Axis: This axis represents the topics (such as bullying sub-themes like ‘body shaming’, ‘racial abuse’, etc.). Y-Axis: This axis represents the sentiment score (negative to positive). Insights: Sentiment explained in relation with each topic carried in this study. For example, in certain cases, topic may be only seen as negative which is useful in determining the extent of damaging topics.

Filtered Data Table Template Purpose: Figures out resolved tweets, their prediction of the category, sentiment scores and topics that were provided. Form: Lines – texts of the up-to-date tweets, the predicted category, the sentiment score and the topics. Filter Options: Users are allowed to filter within the categories, their respective keywords and type of sentiment. Insights: This allows for focused analysis of narrowed down tweets; for example, those labelled hate speech that contained negative sentiments.

Testing

Along with these methods, we accomplished extensive testing in our project which aims to tackle issues related to tweet sentiment analysis and topics modelling in order to assess the complying conducting system performance and reliability. The approach involved the use of various machine learning models for classification purposes with emphasis on the use of sentiments text analysis models like text blob, VADER and transformers from hugging face, plus a neat and even deeply hand assembled hybrid model of Convolutional and Long Short Term Memory when classifying the tweets. These models were assessed using several performance metrics to certify that they are efficient in processing and analyzing tweets in the right manner.

Accuracy is one of the primary metrics that we used in measuring the effectiveness of the performance models pertaining to the sentiment analysis of the tweets and it is focused on assessment of how well the model classified the tweets into predetermined sentiments positive, negative and neutral. Quality of the predictions concerning sentiments was assessed using precision and recall, and F1 score as these metrics give an explanation above the surfaces of the model performance which is important for such models that are likely trained on datasets that are not balanced. It is understood that a high value of the F1 score would mean there is a good between the two variables which is fundamentally important for the trust of the sentiments produced by the model.

In terms of topic modelling we used the Latent Dirichlet Allocation (LDA) model to assign different topics to the tweets. The performance of the model was assessed with a coherence score that reflects how interpretable are the topics identified by the model. The higher the score on coherence the more appropriate the topics are likely to be in relation to the tweets. Moreover, we used perplexity as another supplementary measure to determine the tendency of the model to overfit the data, where lower values indicate less overfitting.

In addition, performance was also on emphasis since we wanted to ensure that the models could cater for large sets of data.

4.3 Significance and advantages of our approach

Our methodology provides a solid basis for carrying out an all-encompassing analysis of social networks through the application of both sentiment analysis and topic modelling techniques. This two-pronged strategy not only assesses social media content regarding the emotions expressed in tweets (positive, negative, neutral), but also describes the major themes which are of interest to the audience. These deposits of sentiment and themes enrich the conversation, creating a more detailed picture of social media activities, thus they are greatly sought in fields professionals such as marketing, public relations, and politics where the understanding of both harness and how the discussions online are directed is paramount. The addition of data collection and processing on a real time basis is yet another pillar of our strategy. Thanks to the system, the user can select specific profile pages and process all gathered tweets without any losses in time. This is of utmost importance in situations where some action must be taken as quickly as possible such as in the case when an event takes place and there is a need to gauge the audience's reactions or feelings about that event, for example marketing or in crisis communication. This ‘big data’ technique eliminates requirements for large organizations and other firms to stockpile vast repertories of static opinions, hence they are able to offer real time evaluation of opinions and attitudes thereby allowing for swift action.

The employment of an array of sentiment analysis models such as TextBlob, VADER, and Hugging Face models stands out as one of the strengths of our approach. These models perform well in varying scenarios with VADER being suitable for short informal text while more complex models such as Hugging Face models help in understanding complex emotions. The addition of these models to the system, improves the performance of sentiment classification and makes it applicable to various genres of twitter content. The use of these models allows the system to maintain a high level of accuracy irrespective of changes in the content and context and hence enhances the system's reliability in carrying out sentimental analyses.

Social Media Content Analysis

Our system combines sentiment analysis and topic modelling to provide a deeper understanding of social media content, focusing on both the emotions and themes in the data..This dual approach is valuable for industries that need detailed insights before making informed decisions, as it captures both the emotional tone and the underlying topics of discussions

Data Collection and Processing in Real-Time

The system is designed to scrape tweets quickly, providing real-time insights in just a few minutes. This capability is crucial for scenarios like ongoing events, crises, or product launches, where timely data can help make decisions based on current public opinion and trends.

Use of Multiple Sentiment Analysis Models

By integrating several sentiment analysis models, the system improves the accuracy and adaptability of sentiment classification. This multi-model approach ensures better results across various types of tweets, including more complex or ambiguous content that traditional systems might struggle to process.

Advanced Topic Modeling with LDA

The system uses Latent Dirichlet Allocation (LDA) for automatic topic modeling, grouping tweets based on shared themes. This method is particularly useful for tracking social media trends and analyzing large datasets, making it easier to identify emerging topics or new ideas that would be too time-consuming to process manually.

CHAPTER 5

CONCLUSION AND FUTURE ENHANCEMENT

In summary, this research has made an effective use of models for advanced machine learning sentiment analysis and topic modelling with the view of extracting useful information available on social media platforms. Incorporating an array of techniques including CNN + LSTM for classification, LDA for topics creation and implementing on a virtual room Streamlit dashboard, The tools will enable the provision of interactive and visualized live data capable of tracking public sentiment, monitoring brands, as well as social media landscapes. This paradigm is strong enough to be extended and used in actual systems for a number of purposes, for example to perform market research or to gauge political opinion.

The use of social sentiment analysis, topic modelling and further categorization using deep learning models avails the robustness of the system. The system also has its scalable system structure and large volume of data processing efficiency capabilities which constitutes great usefulness to the users of fields such as marketing, public relations and social research. By balancing the qualitative and quantitative factors of the social media information and patterns, it becomes possible to create a more elaborate tool for the engagement in discourse over the internet, which will be beneficial for decision making purposes in many areas.

During the course of the project, one or two challenges were faced and all of them were successfully addressed. One of the main issues that had to be dealt with was the precision and the performance of the existing models for sentiment analysis. This is due to the fact that text on social networks contains quite a lot of slang, abbreviations, and phrases which are difficult to interpret outside of their context. In this regard, sentiment analysis was done using different techniques, among them TextBlob, VADER and Hugging Face, which results content was compared for accuracy improvement. From another perspective, there was the challenge of dealing with large amounts of information especially during the scraping as well as preprocessing stage. We improved the way data was handled as well as stored in the system so that up to one hundred thousand tweets can be processed and stored in the system with no degradation in performance. Moreover, the need to combine different machine learning models (CNN + LSTM) for classification posed a challenge as a lot of fine tuning had to be done in order to achieve good performance on classifying tweets, which was done through thorough training and testing on sufficiently large labelled datasets. It is that taking these obstacles into consideration helped build a system of good quality and great reliability. **Multilingual Support:**Adding multilingual support would enable the system to analyze social media data in various languages. This enhancement would provide the ability to perform sentiment analysis and topic modelling on non-English content, expanding the system's utility.

Improved Dashboard Visualization Features: Enhancing the dashboard with more interactive and advanced visualization tools such as heatmaps, word clouds, and time-series graphs would provide users with richer insights. These features would allow for more dynamic data exploration, helping users identify trends and patterns quickly and intuitively.

Automated Report Generation:Implementing automated report generation would allow users to generate detailed reports directly from the system, in formats like PDF or Word. This feature would save time by automating the process of compiling and exporting analysis results..

CHAPTER 6

APPENDICES

Selenium initiation

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from time import sleep
import csv
username = input("Enter your username: ")
profile_url = input("Enter the profile URL of the account: ")
driver = webdriver.Chrome()
driver.get("https://twitter.com/login")
def monitor_tweets(username, profile_url, scroll_lim=5):
    driver.get(profile_url)
    with open(f"{username}_tweets.csv", mode="w", newline="", encoding="utf-8") as file:
        writer = csv.writer(file)
        writer.writerow(["Tweet"])
        for i in range(scroll_lim):
            driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
            sleep(3)
            tweets = driver.find_elements(By.CSS_SELECTOR,
            "div[lang]") for tweet in tweets:
                tweet_text = tweet.text
                writer.writerow([tweet_text])
                print(tweet_text)
            sleep(30)
monitor_tweets(username,
profile_url)
```

LSTM+CNN

```
import pandas as pd
import re
from sklearn.model_selection import train_test_split
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from sklearn.preprocessing import LabelEncoder
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Embedding, Conv1D, MaxPooling1D, LSTM, Dense,
Dropout
def clean_text(text):
    text = str(text) if pd.notnull(text) else ""
    text = re.sub(r'http\S+', '', text)
    text = re.sub(r'@\w+', '', text)
    text = re.sub(r'#\w+', '', text)
    text = re.sub(r'\W', ' ', text)
    return text.lower().strip()
df = pd.read_csv('tweet_labels.csv')
df['tweet'] = df['tweet'].apply(clean_text)
labels = ['Cyberbullying', 'Hate Speech', 'Offensive Language', 'Positive Sentiment',
          'Negative Sentiment', 'Neutral Sentiment', 'Personal Attacks',
          'Harassment', 'Abusive Language', 'Other']
label_encoder = LabelEncoder()
label_encoder.fit(labels)
df['label'] = label_encoder.transform(df['label'])
X_train, X_test, y_train, y_test = train_test_split(df['tweet'], df['label'], test_size=0.2,
random_state=42)
tokenizer =
Tokenizer(num_words=5000)
tokenizer.fit_on_texts(X_train)
X_train_seq = pad_sequences(tokenizer.texts_to_sequences(X_train), maxlen=100)
X_test_seq = pad_sequences(tokenizer.texts_to_sequences(X_test), maxlen=100)
num_classes = len(labels)
model = Sequential()
model.add(Embedding(input_dim=5000, output_dim=128, input_length=100))
model.add(Conv1D(64, kernel_size=3, activation='relu'))
model.add(MaxPooling1D(pool_size=2))
model.add(LSTM(128))
model.add(Dropout(0.5))
model.add(Dense(64,
activation='relu'))
model.add(Dense(num_classes, activation='softmax'))
model.compile(optimizer='adam',
loss='sparse_categorical_crossentropy', metrics=['accuracy'])
history = model.fit(X_train_seq, y_train, epochs=100, batch_size=32,
validation_data=(X_test_seq, y_test))
new_data = pd.read_csv('cleaned_tweets.csv')
new_data['tweet'] = new_data['tweet'].apply(clean_text)
new_data_seq = pad_sequences(tokenizer.texts_to_sequences(new_data['tweet']),
maxlen=100)
predictions = model.predict(new_data_seq)
predicted_labels = predictions.argmax(axis=1)
new_data['label'] =
label_encoder.inverse_transform(predicted_labels)
new_data.to_csv('predicted_tweets.csv', index=False)
```

Sentiment Analysis

```
import os
os.environ["TF_ENABLE_ONEDNN_OPTS"] = "0"
import tensorflow as tf
import pandas as pd
from textblob import TextBlob
from vaderSentiment.vaderSentiment import
SentimentIntensityAnalyzer from transformers import pipeline
sentiment_model = pipeline("sentiment-analysis",
model="distilbert-base-uncased-finetuned-sst-2-english"
) df = pd.read_csv('predicted_tweets.csv')
analyzer = SentimentIntensityAnalyzer()
sentiment_model = pipeline("sentiment-analysis")
def get_textblob_sentiment(text):
  return TextBlob(text).sentiment.polarity
def get_vader_sentiment(text):
    return analyzer.polarity_scores(text)['compound']
def get_transformer_sentiment(text):
    return
    sentiment_model(text)[0]['label']
df['TextBlob_Sentiment'] = df['tweet'].apply(get_textblob_sentiment)
df['VADER_Sentiment'] = df['tweet'].apply(get_vader_sentiment)
df['Transformer_Sentiment'] = df['tweet'].apply(get_transformer_sentiment)
def categorize_sentiment(score):
    if score > 0.05:
        return "Positive"
    elif score < -0.05:
        return
        "Negative"
    else:
        return "Neutral"
df['Sentiment_Category'] = df['VADER_Sentiment'].apply(categorize_sentiment)
df.to_csv('tweets_with_sentiment.csv', index=False)
```

Topic Analysis

```
import pandas as pd
import gensim
from gensim import corpora
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
import nltk
nltk.download('punkt')
nltk.download('stopwords')
df = pd.read_csv('tweets_with_sentiment.csv')
stop_words = set(stopwords.words('english'))
def preprocess_text(text):
    tokens = word_tokenize(text.lower())
    tokens = [word for word in tokens if word.isalpha() and word not in stop_words]
    return tokens
df['Tokens'] =
df['tweet'].apply(preprocess_text) del
dictionary = corpora.Dictionary(df['Tokens']
 corpus = [dictionary.doc2bow(text) for text in df['Tokens']]
num_topics = 5
   lda_model = gensim.models.LdaModel(corpus, num_topics=num_topics,
   id2word=dictionary, passes=10, random_state=42)
   def get_dominant_topic(bow):
topic_scores = lda_model.get_document_topics(bow)
   dominant_topic = max(topic_scores, key=lambda x: x[1])[0]

      return dominant_topic

   df['Dominant_Topic'] = df['Tokens'].apply(lambda
   x: get_dominant_topic(dictionary.doc2bow(x)))
   df.to_csv('tweets_with_topics.csv', index=False)
   print("Topic modeling complete. Output saved to 'tweets_with_topics.csv'.")
```

REFERENCES

[1] Mitushi Raj, Samridhi Singh, Kanishka Solanki, Ramani Selvanambi (2022). “An application to detect cyberbullying using machine learning and deep learning techniques”. SN Computer Science,Volume:3. https://doi.org/10.1007/s42979-022-01308-5

[2] Aditya Desai,Shashank Kalaskar,Omkar Kumbhar, Rashmi Dhumal (2021).“Cyber Bullying Detection on Social Media using Machine Learning”. ITM Web of Conferences, Volume:40, 03038. https://doi.org/10.1051/itmconf/20214003038

[3] Aljwharah Alabdulwahab, Mohd Anul Haq, Mohammed Alshehri, (2023). “Cyberbullying Detection using Machine Learning and Deep Learning”. International Journal of Advanced Computer Science and Applications,Volume:14,Issue:10. https://doi.org/10.14569/ijacsa.2023.0141045

[4] Miss. Jafri Sayeedaaliza Abutorab, Miss. Wagh Roshani Balasaheb, Miss. Gaikwad Vaishnavi Subodh, Miss. Sonawane Ujjwala Dattu, Professor Waghmare.A.I.“DETECTION OF CYBERBULLYING ON SOCIAL MEDIA USING MACHINE LEARNING”. (2022). International Research Journal of Modernization in Engineering Technology and Science, Volume:04, e-ISSN:2582–5208, https://www.irjmets.com/uploadedfiles/paper/issue_5_may_2022/24749/final/fin_irjmets1653789970.pdf

[5] John Hani Mounir, Mohamed Nashaat, Mostafaa Ahmed, Zeyad Emad, Eslam Amer, Ammar Mohammed, (2019). “Social Media Cyberbullying Detection using Machine Learning”. International Journal of Advanced Computer Science and Applications, Volume:10, Issue:5 . https://doi.org/10.14569/ijacsa.2019.0100587

[6] Daniyar Sultan, AigerimToktarova, Ainur Zhumadillayeva, Sapargali Aldeshov, Shynar Mussiraliyeva, (2022). “Cyberbullying-related hate speech detection using shallow-to-deep learning”. Computers, Materials & Continua/Computers, Materials & Continua (Print),Volume: 74, e-ISSN:2115–2131. https://doi.org/10.32604/cmc.2023.032993

[7] Raju Kumar, Aruna Bhat (2022).” A study of machine learning-based models for detection, control, and mitigation of cyberbullying in online social media.” International Journal of Information Security, Volume:21,Issue:6, 1409–1431.

[8] Edla Hareen (2022). “CYBERBULLYING DETECTION IN SOCIAL NETWORK” [Research article].International Journal for Research Trends and Innovation,Volume:7,Issue:12,ISSN:2456-3315. https://www.ijrti.org/papers/IJRTI2212054.pdf

[9] Ravindra Chilbule, Kasifraza Siddique, Sangharsh Moon, Nirmal Zade, Aditya Fusate. (2023). "Cyber bullying and hate speech detection [Journal-article]". International Journal of Advanced Research in Science, Communication and Technology, Volume:3, Issue:12. https://doi.org/10.48175/IJARSCT-10709

[10] Sneha Gajanan Sambare, L.Haridas, Dr.Sanjay (2022)." A review paper on Cyber Harassment Detection using Machine learning Algorithm on social networking website." International Journal for Research in Applied Science and Engineering Technology, Volume:10, Issue:10, pp:780–785. https://doi.org/10.22214/ijraset.2022.45847

[11] Cinare Oguz Aliyeva, Mete Yaganoglu(2024)." Deep learning approach to detect cyberbullying on twitter." Multimedia Tools and Applications. https://doi.org/10.1007/s11042-024-19869-3

[12] Logasree.S, Harshini.M, Arasu college of arts and science for women. (2023). "Cyberbullying Detection using machine learning" ,Volume:05, Issue:04, pp:124–125. https://www.irjweb.com/Cyberbullying%20Detection%20using%20machine%20learning.pdf

[13] Neelakandan S, Sridevi M, Saravanan Chandrasekaran, Murugeswari K, Aditya Kumar Singh Pundir, Sridevi R,T.Bheema Lingaiah,(2022)."Deep learning approaches for cyberbullying detection and classification on social media." Computational Intelligence and Neuroscience, 2022, Issue:1–13. https://doi.org/10.1155/2022/2163458

[14] Dong-Hwi Kim,Woo-Hyeok Son,Sung-Shin Kwak,Tae-Hyeon Yun,Ji-Hyeok Park andJae-Dong Lee (2023). "A hybrid deep learning emotion classification system using multimodal data. Sensors" Volume:23, Issue:23, 9333. https://doi.org/10.3390/s23239333

[15] Md Saroar Jahan, Mourad Oussalah (2023). "A systematic review of hate speech automatic detection using natural language processing." Neurocomputing, Volume:546, 126232. https://doi.org/10.1016/j.neucom.2023.126232

[16] Malliga Subramanian, Veerappampalayam Easwaramoorthy Sathiskumar, G. Deepalakshmi, Jaehyuk Cho, G. Manikandan (2023). "A survey on hate speech detection and sentiment analysis using machine learning and deep learning models." Alexandria Engineering Journal,Volume: 80, Pages:110–121. https://doi.org/10.1016/j.aej.2023.08.038

[17] Donia Gamal 1,,Marco Alfonse ,Salud María Jiménez-Zafra, Mostafa Aref (2023). "Intelligent Multi-Lingual Cyber-Hate Detection in Online Social Networks: Taxonomy, approaches, datasets, and open challenges". Big Data and Cognitive Computing, Volume: 7, Issue:2. https://doi.org/10.3390/bdcc7020058

[18] Douglas C Youvan (2024). “AI Detection of Subtle Hate Speech in Social Media.http://dx.doi.org/10.13140/RG.2.2.19317.59362

[19] Seble.H, Muluken.S, Edemealem.D, Kafte.T, Terefe.F, Mekashaw.G, Abiyot.B, Senait.T. “HATE SPEECH DETECTION USING MACHINE LEARNING: A SURVEY.” Academy Journal of Science and Engineering, Volume:17. https:// www.researchgate .net/ publication/374295515_HATE_SPEECH_DETECTION_USING_MACHINE_LEARNING _A_SURVEY

[20] Md. Tarek Hasan,Md. Al Emran Hossain ,Md. Saddam Hossain Mukta ,Arifa Akter,Mohiuddin Ahmed andSalekul Islam (2023). “A review on Deep-Learning-Based Cyberbullying Detection.”Future Internet, Volume:15, Issue:5, 179. https://doi.org/10.3390/fi15050179

[21] Dr. Vijayakumar V,Dr Hari Prasad D(2021).”A STUDY ON DEEP LEARNING ALGORITHMS FOR MULTIMODAL AND MULTILINGUAL CYBERBULLYING DETECTION”, INDIAN JOURNAL OF APPLIED RESEARCH, Volume - 11, Issue - 07 https://doi.org/10.36106/ijar

[22] ZAINAB MANSUR, NAZLIA OMAR,SABRINA TIUN (2023).” Twitter Hate Speech Detection: A Systematic Review of Methods, Taxonomy Analysis, Challenges, and Opportunities”.IEEEAccess,Volume-11,Pages-16226–16249.https://doi.org/10.1109/access. 2 023.3239375

[23] Miriam Di Lisio, Rosa Sorrentino, Domenico Trezza & University of Naples Federico II. (2022).” Platformization hate. Patterns and algorithmic bias of verbal violence on social media”.In Mediascapes Journal, Volume-20. https://rosa.uniroma1.it/rosa03/mediascapes/article/download/18039/17287/38009